

**ANÁLISIS, DISEÑO Y DESARROLLO DE UN PROTOTIPO DE SOFTWARE PARA LA
ADMINISTRACIÓN DE PARQUEADEROS**

FRANK SILVA CASTRO

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
FACULTAD DE INGENIERÍAS
AGOSTO 2016**

**ANÁLISIS, DISEÑO Y DESARROLLO DE UN PROTOTIPO DE SOFTWARE PARA LA
ADMINISTRACIÓN DE PARQUEADEROS**

*Memoria que se presenta para optar al título de
Ingeniero de Sistemas y Computación*

FRANK SILVA CASTRO

Director de proyecto

PhD. GUILLERMO ROBERTO SOLARTE MARTINEZ

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
FACULTAD DE INGENIERÍAS
AGOSTO 2016**

*A mi madre, que siempre ha creído en mí, que ha trabajado más que
nadie por este título, incluso más que yo.*

A mi padre, que sin darse cuenta me animó a ser alguien en la vida.

Ambos fuente de inspiración desde mi primer día y hasta el último.

Por último, a mis profesores, en especial a mi director de proyecto.

Gracias a ustedes y sus enseñanzas puedo considerarme un profesional.

Frank Silva Castro

Agradecimientos

A mi familia, amigos, compañeros, profesores y conocidos; que siempre creyeron en mí. Son ustedes los que hicieron este título posible. A los profesores por compartir sus conocimientos y enseñarme todo lo que conozco hoy en día, directa o indirectamente.

A mi director de proyecto, por asesorarme y acompañarme a través de la realización de este documento.

Frank Silva Castro

Resumen

Este proyecto nace a partir de la necesidad de un software que facilite la administración de un establecimiento dedicado al alquiler de parqueaderos, de modo que muchas de las características de estos establecimientos sean parametrizables desde una herramienta de software.

En el proyecto, se diseñó y se desarrolló una aplicación de escritorio, compatible con versiones de Windows (7, 8, 8.1, 10) que permite la administración, parametrización y ejecución de las tareas realizadas en un establecimiento de alquiler de parqueaderos, con características extendidas como préstamo de inventario (cascos, chalecos) y creación de cupos de parqueo.

El prototipo se desarrolló en la plataforma Windows para aprovechar la infraestructura básica de un parqueadero de la ciudad de Pereira.

La concepción del software incluye actividades como; gestión de clientes, empleados, quejas, vehículos, mensualidades, ingresos, egresos, bitácora, auditoría, inventario, préstamos y facturación.

Para el desarrollo del prototipo se utilizó una arquitectura cliente/servidor, de modo que los diferentes equipos estén conectados a una misma base de datos, compartiendo entre sí cada gestión. Se utilizó el IDE WINDEV y el lenguaje WLANGUAGE para el desarrollo.

Abstract

This Project was born because of the needing of a software that simplifies the administration of a parking establishment, in a way that many of the characteristics of this establishments could be set from a software tool.

In this project, have been designed and developed a desktop app, that works on some Windows versions (7, 8, 8.1, 10), and allows the administration, setting up, and execution of the tasks performed in a parking establishment, with extended characteristics like inventory lending (helmets, etc.) and the creation of parking sites.

The prototype has been developed in the Windows platform to take advantage of the basic infrastructure of a parking in Pereira city.

The software conception includes activities like: client management, employees, complaints, vehicles, monthly installments, incomes, expenses, binnacle, audit, inventory, lending, and billing.

In order to develop the prototype, has been used a client/server architecture, in order to the different PCs where connected to a same database, sharing between then every management. Has been used the IDE WINDEV, and the language WLANGUAGE for the development.

Índice

Título.....	1
Plan fundamental	2
Descripción del problema	2
Planteamiento del problema	2
• FORMULACIÓN	2
• SISTEMATIZACIÓN	3
Objetivos	3
Objetivo general.....	3
Objetivos específicos	3
Justificación	3
Plan procedimental	4
Descripción del proyecto	4
Descripción del sistema.....	6
Recursos	6
Recursos humanos	6
Recursos técnicos y tecnológicos	6
Metodología.....	7
Cronograma	8
Cronograma detallado	8
Análisis del sistema	10
Historias de usuario	11
EMPLEADOS	11
ADJUNTOS	11
QUEJAS	12
INGRESOS Y EGRESOS	13
CONFIGURACIÓN	13
BITÁCORA	14
INVENTARIOS	14
CLIENTES	15

VEHÍCULOS	15
MENSUALIDADES	16
FACTURACIÓN	16
PRÉSTAMOS	17
ESTADÍSTICAS	17
Otras Interfaces	18
Requerimientos de software	19
Casos de uso	20
Descripción de casos de uso	21
• Get_All_x	21
• CreMod_x	21
• Eliminar_x	21
Diagramas de secuencia	73
Creación	73
Eliminar	80
Modificar	86
Ver	91
Diseño del sistema.....	97
Diseño arquitectónico	97
Diagrama de clases.....	98
Tipos de función.....	98
• Get_All_x	98
• CreMod_x	98
• Eliminar_x	99
Descripción de diagramas de clase	99
ADJUNTO	99
AUDITORÍA	99
BITÁCORA	99
BLOQUE	100
BODEGA	100
CLIENTE.....	101
CONFIGURACIÓN	101
CUPO	101

EMPLEADO	102
INGRESO	102
INVENTARIO	103
LÍNEA	103
MENSUALIDAD	103
PRÉSTAMO.....	104
QUEJA.....	104
VEHÍCULO	105
VOUCHER.....	105
Usuarios	106
Perfiles de usuario	106
Interfaces.....	107
Módulo de quejas.....	107
Módulo de auditoría	108
Módulo de inventario.....	109
Módulo de bitácora	110
Módulo de bodega	111
Módulo de ingresos y egresos	112
Módulo de clientes	113
Módulo de vehículos	114
Módulo de bloques.....	115
Módulo de préstamos	116
Módulo de mensualidades	117
Módulo de facturas	118
Conclusiones	119
Director de trabajo	121
Colaboradores	121
Bibliografía	122
Capacitación.....	122

Índice de ilustraciones

Ilustración 1: diagrama de casos de uso	20
Ilustración 2: crear configuración	73
Ilustración 3: crear adjunto	73
Ilustración 4: crear bitácora	74
Ilustración 5: crear bloque	74
Ilustración 6: crear bodega	75
Ilustración 7: crear cupo	75
Ilustración 8: crear empleado	76
Ilustración 9: crear ingreso	76
Ilustración 10: crear inventario	77
Ilustración 11: crear mensualidad	77
Ilustración 12: crear préstamo	78
Ilustración 13: crear queja	78
Ilustración 14: crear vehículo	79
Ilustración 15: crear voucher	79
Ilustración 16: eliminar adjunto	80
Ilustración 17: eliminar bitácora	80
Ilustración 18: eliminar bodega	81
Ilustración 19: eliminar empleado	81
Ilustración 20: eliminar ingreso	82
Ilustración 21: eliminar inventario	82
Ilustración 22: eliminar línea	83
Ilustración 23: eliminar mensualidad	83
Ilustración 24: eliminar préstamo	84
Ilustración 25: eliminar queja	84
Ilustración 26: eliminar vehículo	85
Ilustración 27: eliminar voucher	85
Ilustración 28: modificar adjunto	86
Ilustración 29: modificar bitácora	86
Ilustración 30: modificar bloque	87
Ilustración 31: modificar cupo	88
Ilustración 32: modificar empleado	88
Ilustración 33: modificar inventario	89
Ilustración 34: modificar línea	89
Ilustración 35: modificar mensualidad	90
Ilustración 36: modificar queja	90
Ilustración 37: modificar vehículo	91
Ilustración 38: ver adjunto	91
Ilustración 39: ver auditoría	92
Ilustración 40: ver bitácora	92
Ilustración 41: ver bloque	92

Ilustración 42: ver bodega	93
Ilustración 43: ver cupo	93
Ilustración 44: ver empleado	93
Ilustración 45: ver ingreso	94
Ilustración 46: ver inventario	94
Ilustración 47: ver línea	94
Ilustración 48: ver mensualidad.....	95
Ilustración 49: ver préstamo	95
Ilustración 50: ver queja	95
Ilustración 51: ver vehículo	96
Ilustración 52: ver voucher	96
Ilustración 53: diagrama de clases	98
Ilustración 54: módulo de quejas.....	107
Ilustración 55: módulo de auditoría	108
Ilustración 56: módulo de inventario.....	109
Ilustración 57: módulo de bitácora	110
Ilustración 58: módulo de bodega	111
Ilustración 59: módulo de ingresos y egresos	112
Ilustración 60: módulo de clientes	113
Ilustración 61: módulo de vehículos	114
Ilustración 62: módulo de bloques.....	115
Ilustración 63: módulo de préstamos	116
Ilustración 64: módulo de mensualidades	117
Ilustración 65: módulo de facturas	118

Índice de tablas

Tabla 1: uso de versiones de Windows.....	4
Tabla 2: cronograma de trabajo	8
Tabla 3: cronograma detallado de trabajo.....	8
Tabla 4: formulario de creación/modificación de empleados.....	11
Tabla 5: formulario de CRUD de adjuntos.....	11
Tabla 6: formulario de CRUD de quejas.....	12
Tabla 7: formulario de resolución de quejas.....	12
Tabla 8: registro de ingresos y egresos	13
Tabla 9: configuración de parámetros generales	13
Tabla 10: Formulario de bitácora	14
Tabla 11: formulario de inventarios	14
Tabla 12: formulario de clientes	15
Tabla 13: formulario de vehículos	15
Tabla 14: formulario de mensualidades	16
Tabla 15: formulario de facturación.....	16
Tabla 16: formulario de préstamo	17
Tabla 17: estadísticas de ingresos y egresos	17
Tabla 18: auditoría	18
Tabla 19: requerimientos de software	19
Tabla 20: ver adjuntos.....	21
Tabla 21: crear adjunto	22
Tabla 22: modificar adjunto	23
Tabla 23: eliminar adjunto.....	23
Tabla 24: ver auditoría	24
Tabla 25: ver bitácora	25
Tabla 26: crear bitácora	26
Tabla 27: modificar bitácora.....	27
Tabla 28: eliminar bitácora.....	28
Tabla 29: ver bloques.....	28
Tabla 30: crear bloques	29
Tabla 31: modificar bloque.....	30
Tabla 32: eliminar bloque.....	31
Tabla 33: ver bodegas	32
Tabla 34: crear bodega	32
Tabla 35: modificar bodega.....	33
Tabla 36: eliminar bodega.....	34
Tabla 37: ver cliente.....	35
Tabla 38: crear cliente.....	36
Tabla 39: modificar cliente	37
Tabla 40: eliminar cliente	37
Tabla 41: ver configuración	38

Tabla 42: crear configuración	39
Tabla 43: ver cupo	40
Tabla 44: crear cupo	40
Tabla 45: modificar cupo	41
Tabla 46: eliminar cupo	42
Tabla 47: ver empleados	43
Tabla 48: crear empleado	44
Tabla 49: modificar empleado	45
Tabla 50: eliminar empleado	45
Tabla 51: ver ingreso	46
Tabla 52: crear ingreso o egreso	47
Tabla 53: eliminar ingreso o egreso	48
Tabla 54: ver inventario	49
Tabla 55: crear inventario	50
Tabla 56: modificar inventario	50
Tabla 57: eliminar inventario	51
Tabla 58: ver línea	52
Tabla 59: crear línea	53
Tabla 60: modificar línea	54
Tabla 61: eliminar línea	55
Tabla 62: ver mensualidad	56
Tabla 63: crear mensualidad	56
Tabla 64: modificar mensualidad	57
Tabla 65: eliminar mensualidad	58
Tabla 66: ver préstamo	59
Tabla 67: crear préstamo	60
Tabla 68: modificar préstamo	61
Tabla 69: eliminar préstamo	61
Tabla 70: ver queja	62
Tabla 71: crear queja	63
Tabla 72: modificar queja	64
Tabla 73: eliminar queja	65
Tabla 74: ver vehículo	66
Tabla 75: crear vehículo	66
Tabla 76: modificar vehículo	67
Tabla 77: eliminar vehículo	68
Tabla 78: ver voucher	69
Tabla 79: crear voucher	70
Tabla 80: modificar voucher	71
Tabla 81: eliminar voucher	72
Tabla 82: perfiles de usuarios	106

Capítulo 1

Título

Análisis, diseño y desarrollo de un prototipo de software para la administración de parqueaderos.

Capítulo 2

Plan fundamental

Descripción del problema

Este proyecto se desarrolla en la Universidad Tecnológica de Pereira, ubicada en el suroriente de la ciudad de Pereira, en Risaralda (dentro del área metropolitana de centro occidente). Los establecimientos de alquiler de parqueaderos generalmente no cuentan con un sistema de administración, lo que hace que la gestión de ingresos, dinero, clientes y otras actividades se vuelvan un tanto engorrosas. Se propone el análisis, diseño y desarrollo de un prototipo de software para la gestión de parqueaderos.

Planteamiento del problema

En Bogotá, el 23 de enero de 2015, existen aproximadamente entre 1.700 y 3.000 parqueaderos¹

Extendiendo la cifra, podemos asumir que, en Colombia, hay aproximadamente 10.000 parqueaderos, siendo ésta una cifra pequeña.

La administración de un parqueadero contempla varios aspectos, como lo son; la facturación, gestión de mensualidades, verificación y gestión de cupos, e informes, entre otros aspectos.

En muchos de los establecimientos, este proceso se realiza manualmente. Lo que supone una pérdida de tiempo considerable, que afecta la ejecución de las tareas del administrador.

Se pretende implementar un software que facilite la administración de un parqueadero; que permita la facturación, la asignación automática de cupos, verificación de cupos restantes, generación de reportes, informes, y el control de mensualidades, de modo que las realizaciones de las tareas del administrador del establecimiento sean asistidas por medio del software mencionado, permitiendo un ahorro de tiempo considerable en sus tareas.

Se pretende, además, crear módulos adicionales que se encarguen de manejar la concurrencia en cupos de parqueo.

● FORMULACIÓN

¿Es posible implementar un software que se encargue de servir como medio para administrar un establecimiento de parqueo, asistiendo las funciones de facturación, cupos, informes, y gestión de cupos con manejo de concurrencias?

¹DIARIO LA REPÚBLICA [www.larepublica.co]. 2016. Secretaría tiene bajo la lupa 3000 parqueaderos de Bogotá, [consulta 24 Agosto 2016]. Disponible en: http://www.larepublica.co/secretar%C3%ADa-tiene-bajo-la-lupa-3000-parqueaderos-de-bogot%C3%A1_211946

● SISTEMATIZACIÓN

- ¿Qué tecnologías disponibles se pueden utilizar?
- ¿Cómo se van a integrar estas tecnologías?
- ¿En qué plataforma y bajo qué requisitos se implementará el software?

Objetivos

Objetivo general

Desarrollar un prototipo de software que permita la gestión de un parqueadero, asistiendo en tareas como: facturación, generación de informes, manejo de mensualidades, y gestión de cupos con manejo de concurrencias.

Objetivos específicos

1. Recolectar información acerca del funcionamiento de un parqueadero; su modo de facturación, gestión de cupos, manejo de mensualidades, y otros aspectos necesarios.
2. Definir los requerimientos de software y hardware para la implementación del prototipo.
3. Realizar las etapas de análisis y diseño para el prototipo a desarrollar aplicando las técnicas de ingeniería del software.
4. Desarrollar el prototipo de software para la administración de parqueaderos.
5. Realizar pruebas sobre el software desarrollado al interior de la UTP.

Justificación

Las herramientas de software son concebidas con el fin de dar solución a un problema o agilizar un proceso. Es necesario aplicar el uso de software en cualquier ámbito; las tareas a realizar se facilitan a través de éstos, y esto mismo aplica directamente al caso de estudio: los establecimientos de alquiler de parqueadero.

Dentro de algunos parqueaderos existen herramientas de software utilizadas para mejorar la gestión del establecimiento, herramientas dedicadas a tareas simples como la facturación, o gestión de clientes. Por estas razones es necesario que se implemente una solución completa, que tenga en cuenta muchas de las actividades que se realizan dentro de este tipo de establecimientos; gestión de ingresos, egresos, clientes, vehículos, mensualidades, empleados, quejas, bitácora, auditoría y parametrización en una sola herramienta.

Capítulo 3

Plan procedimental







Descripción del proyecto

El proyecto descrito busca beneficiar a los administradores de establecimientos de alquiler de parqueadero, inicialmente de la ciudad de Pereira. Se utilizó la plataforma Windows, gracias a su trayectoria histórica y soporte actual, siendo una plataforma muy utilizada en la ciudad de Pereira.

Se desarrolló en el IDE WINDEV, el motor de base de datos HFSQL y el lenguaje de programación WLANGUAGE, eligiendo estas tecnologías gracias al conocimiento y soporte de la herramienta por parte de los autores del proyecto.

A continuación, se muestra la cantidad de usuarios que utilizan las diferentes versiones de Windows².

Tabla 1: uso de versiones de Windows

Fuente	Net Market Share ³	W3Counter ⁴	Global Stats ⁵
Fecha	Mayo de 2016	Mayo de 2016	Mayo de 2016
Total	 89.68%	 38.55%*	 72.27%
Windows 7 (2009)	 48.57%	 21.57%	 38.95%

Fuente: WIKIPEDIA [www.wikipedia.com]. 2016. Versiones más utilizadas, [consulta 24 agosto 2016]. Disponible en: https://es.wikipedia.org/wiki/Microsoft_Windows

² Incluyen sistemas operativos para móviles, y no acepta valores de menos del 3%. Sólo incluye "Windows 8.1", por lo que el resultado podría estar incompleto.

³ HITS LINK NET APPLICATIONS [www.hitslink.com]. 2016. Desktop Operating System Market Share, [consulta 24 Agosto 2016]. Disponible en: <http://marketshare.hitslink.com/operating-system-market-share.aspx?qprid=10&qpcustomd=0&qpsp=175&qpn=1&qptimeframe=M>

⁴ W3COUNTER [www.w3counter.com]. 2016. Browser & Platform Market Share, [consulta 24 Agosto 2016]. Disponible en: <https://www.w3counter.com/globalstats.php?a%C3%B1o=2014&mes=3>

⁵ STATCOUNTER [www.statcounter.com]. 2016. Top 7 Desktop & Console Oss from 5 to 7 Apr 2014, [consulta 24 Agosto 2016]. Disponible en: <http://gs.statcounter.com/#os-ww-daily-20140405-20140407-bar>

Fuente	Net Market Share ⁶	W3Counter ⁷	Global Stats ⁸
Windows 10 (2015)	▲ 17.43%	▲ 9.18%	▲ 19.33%
Windows 8.x (2012)	▼ 11.39%	▼ 4.78%**	▼ 8.08%**
Windows XP (2001)	▼ 10.09%	▼ 3.02%	▼ 5.91%
Windows Vista (2007)	▼ 1.35%		
Windows Server (2000)	▲ 0.75%		
Windows 3.1 (1990)	▼ 0.10%		

Fuente: WIKIPEDIA [www.wikipedia.com]. 2016. Versiones más utilizadas, [consulta 24 agosto 2016]. Disponible en: https://es.wikipedia.org/wiki/Microsoft_Windows

A partir de la tabla anterior se elige como versión base de Windows, la versión Windows 7, ya que es la versión más utilizada por los usuarios, aun así, el software puede ser usado en las demás versiones de Windows, a partir de Windows XP.

⁶ HITS LINK NET APPLICATIONS [www.hitslink.com]. 2016. Desktop Operating System Market Share, [consulta 24 agosto 2016]. Disponible en: <http://marketshare.hitslink.com/operating-system-market-share.aspx?qprid=10&qpcustomd=0&qpsp=175&qpn=1&qptimeframe=M>

⁷ W3COUNTER [www.w3counter.com]. 2016. Browser & Platform Market Share, [consulta 24 agosto 2016]. Disponible en: <https://www.w3counter.com/globalstats.php?a%C3%B1o=2014&mes=3>

⁸ STATCOUNTER [www.statcounter.com]. 2016. Top 7 Desktop & Console Oses from 5 to 7 Apr 2014, [consulta 24 agosto 2016]. Disponible en: <http://gs.statcounter.com/#os-ww-daily-20140405-20140407-bar>

Descripción del sistema

El proyecto se basa en la arquitectura cliente/servidor; con un equipo portátil, de escritorio, o servidor; como equipo servidor, y con equipos portátiles o de escritorio; como clientes livianos.

Los equipos cliente se conectan directamente a la base de datos del equipo servidor, para usar su base de datos remotamente. Para evitar la colisión de datos, y las operaciones cruzadas, se utilizan transacciones para bloquear los registros, y operaciones COMMIT, para aprobar las transacciones.

Un cliente liviano se refiere a que la carga del procesamiento de datos recae principalmente en el equipo servidor; dejando al cliente la tarea de consumir recursos y presentarlos.

Recursos

Recursos humanos

- Frank Silva Castro

Asesor

- PhD. Guillermo Roberto Solarte Martínez

Recursos técnicos y tecnológicos

Recursos técnicos

- IDE WINDEV 20
- Motor de base de datos HFSQL
- SO WINDOWS 10

Recursos tecnológicos

- Equipo de escritorio con sistema operativo Windows 10 y Ubuntu 14
- Tablet Windows 8.1
- Resma de papel
- Tinta de impresora (1 lt)
- Impresora Epson L120
- Fotocopiadora

Metodología

Este proyecto llevará a cabo mediante las siguientes etapas:

Etapas 1. Investigaciones pertinentes.

- Investigar acerca del modo de facturación, gestión de cupos, manejo de mensualidades, y aspectos importantes de la gestión de parqueaderos.

Etapas 2. Selección de la tecnología.

- Se consultará en libros, internet y revistas, las tecnologías disponibles para realizar el desarrollo bajo la plataforma.
- De acuerdo con la consulta se seleccionará la tecnología para realizar el desarrollo del software

Etapas 3. Ingeniería del Software.

- Establecer el funcionamiento básico del software
- Realizar el documento de alcance del software
- Realizar los diagramas del Modelo de Vista Arquitectónica 4+1 de Kruchten

Etapas 4. Desarrollo de Software.

- Construir la base de datos
- Construir el entorno básico de la aplicación
- Construir el módulo creación de cupos (CRUD)
- Construir el módulo creación de clientes y mensualidades (CRUD)
- Construir las funciones de concurrencia y asignación de cupos
- Construir el módulo de facturación

Cronograma

Tabla 2: cronograma de trabajo

DIAGRAMA DE GANTT: CRONOGRAMA DE TRABAJO				
TAREA	DURACIÓN			
	MES 1	MES 2	MES 3	MES 4
Recopilación de datos de los interesados				
Investigación de herramientas, tecnologías, metodologías y estándares				
Levantamiento de requerimientos				
Análisis y diseño de software				
Desarrollo de software				
Pruebas de software				

Ref: Autor

Cronograma detallado

Tabla 3: cronograma detallado de trabajo

N	Actividad	Descripción	Tiempo de ejecución
1	Recopilación de datos	Recopilar datos sobre las personas que son objeto de estudio, para determinar una solución óptima al problema planteado. La recolección puede ser realizada por medio de encuestas físicas o virtuales.	1 Mes
2	Investigar y seleccionar una tecnología de desarrollo	Analizar las diferentes tecnologías de desarrollo existentes para elegir la que más se adapte a las necesidades del desarrollador. El análisis puede ser basado	1 Mes

		sobre consultas en internet y estándares definidos.	
3	Definir los requerimientos de software y hardware para la implementación del prototipo	Analizar los requerimientos de software y hardware necesarios para la implementación del prototipo. El análisis puede ser basado sobre consultas en internet y estándares definidos.	1 Mes
4	Realizar las etapas de análisis y diseño para el prototipo a desarrollar aplicando las técnicas de ingeniería del software	Realizar el análisis de requerimientos y elaborar los diagramas de Krutchen para el prototipo de software a desarrollar.	2 Meses
5	Desarrollar el prototipo de aplicativo WEB para la administración de torneos de fútbol locales	Desarrollar el prototipo de software con respecto a los datos obtenidos, la ingeniería del software realizada y los estándares de desarrollo definidos.	2 Meses
6	Realizar pruebas sobre el software desarrollado.	Realizar las pruebas de software con las técnicas de caja blanca y caja negra. De modo que se pruebe la funcionalidad y calidad del software	1 Mes

Ref: Autor

Capítulo 4

Análisis del sistema

Para este proyecto se hizo uso del patrón de arquitectura Modelo-Vista-Controlador. El patrón MVC define el modelo como el acceso a la base de datos, la vista como las interfaces de usuario o sistema, y el controlador como el que interactúa entre la vista y el modelo, gestionando el flujo de trabajo del proyecto.⁹

⁹ MICROSOFT MSDN [msdn.microsoft.com]. 2016. Model-View-Controller, [consulta 24 agosto 2016]. Disponible en: <https://msdn.microsoft.com/en-us/library/ff649643.aspx>

Historias de usuario

Una historia de usuario es una representación de un requisito escrito en una o dos frases utilizando el lenguaje común del usuario. Las historias de usuario son utilizadas en las metodologías de desarrollo ágiles para la especificación de requisitos (acompañadas de las discusiones con los usuarios y las pruebas de validación).¹⁰

A continuación, se listan las historias de usuario obtenidas para el desarrollo del proyecto, discriminadas por módulos. Se utilizarán las abreviaciones; **A**: Administrador, **E**: Empleado, **CRUD**: Create, Read, Update, Delete, operaciones básicas de bases de datos.

EMPLEADOS

Tabla 4: formulario de creación/modificación de empleados

Historia de Usuario		
Número: 1	Nombre: Formulario de creación y modificación de empleados (A)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador, espero poder registrar nuevos empleados, editarlos, eliminarlos, y restablecer sus contraseñas por medio de correo electrónico.		
Observaciones: Los empleados, por primera vez, sólo pueden ser creados por el usuario administrador		

Ref: Autor

ADJUNTOS

Tabla 5: formulario de CRUD de adjuntos

Historia de Usuario		
Número: 1	Nombre: Formulario de consulta, creación, modificación y eliminación de empleados (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :1	

¹⁰ Daniel H. Steinberg and Daniel W. Palmer: Extreme Software Engineering, Pearson Education, Inc., ISBN 0-13-047381-2

Descripción: Yo, como administrador/empleado, espero poder consultar, crear, modificar, y eliminar imágenes de documentos como facturas, comprobantes, formularios, entre otros; para llevar un registro de los documentos del establecimiento.
Observaciones: Sólo podrán ser cargadas imágenes, no documentos word, excel, o demás

Ref: Autor

QUEJAS

Tabla 6: formulario de CRUD de quejas

Historia de Usuario		
Número: 1	Nombre: Formulario de consulta, creación, modificación y eliminación de quejas (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador/empleado, espero poder consultar, crear, modificar, y eliminar quejas de los clientes en el sistema, acerca del servicio prestado.		
Observaciones: Las quejas creadas no tienen respuesta de resolución, hasta que se le es asignada		

Ref: Autor

Tabla 7: formulario de resolución de quejas

Historia de Usuario		
Número: 2	Nombre: Formulario de resolución de quejas (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador/empleado, espero poder resolver las quejas creadas en el sistema.		
Observaciones: Sin observaciones		

Ref: Autor

INGRESOS Y EGRESOS

Tabla 8: registro de ingresos y egresos

Historia de Usuario		
Número: 1	Nombre: Formulario de registro de ingresos y egresos (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador/empleado, espero poder registrar nuevos ingresos y egresos de dinero manualmente. También se requiere que los ingresos por facturación se registren automáticamente en el sistema		
Observaciones: Los ingresos y egresos creados, sólo pueden ser eliminados por el usuario administrador		

Ref: Autor

CONFIGURACIÓN

Tabla 9: configuración de parámetros generales

Historia de Usuario		
Número: 1	Nombre: Configuración de parámetros (A)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador, espero poder establecer las diferentes tarifas para la facturación dentro del sistema, para que sean usadas automáticamente por el usuario empleado.		
Observaciones: La configuración sólo puede ser modificada por el usuario administrador		

Ref: Autor

BITÁCORA

Tabla 10: Formulario de bitácora

Historia de Usuario		
Número: 1	Nombre: formulario de bitácora (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador/empleado, espero poder comunicarme con los demás empleados a través de una bitácora, donde se escriban notas que los demás empleados puedan leer		
Observaciones: La bitácora no puede ser modificada o eliminada		

Ref: Autor

INVENTARIOS

Tabla 11: formulario de inventarios

Historia de Usuario		
Número: 1	Nombre: formulario de bitácora (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador/empleado, espero poder ingresar o retirar elementos del inventario, cuando sea necesario; para poder realizar préstamos a los clientes, de los elementos creados en inventario		
Observaciones: La cantidad de elementos creados/prestados, no puede ser modificada		

Ref: Autor

CLIENTES

Tabla 12: formulario de clientes

Historia de Usuario		
Número: 1	Nombre: formulario de clientes (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador/empleado, espero poder consultar, crear, modificar y eliminar clientes, para facturar		
Observaciones: Sin observaciones		

Ref: Autor

VEHÍCULOS

Tabla 13: formulario de vehículos

Historia de Usuario		
Número: 1	Nombre: formulario de vehículos (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :1	
Descripción: Yo, como administrador/empleado, espero poder consultar, crear, modificar y eliminar vehículos para prestación de servicios, además de poder relacionarlos con clientes		
Observaciones: Un vehículo puede ser relacionado con un cliente		

Ref: Autor

MENSUALIDADES

Tabla 14: formulario de mensualidades

Historia de Usuario		
Número: 1	Nombre: formulario de mensualidades (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :2	
Descripción: Yo, como administrador/empleado, espero poder inscribir mensualidades con una vigencia de tiempo, para relacionarla con un cliente, y sus vehículos		
Observaciones: La mensualidad debe cargarse automáticamente a la hora de facturar Las mensualidades no pueden borrarse El registro de mensualidad debe generar un ingreso		

Ref: Autor

FACTURACIÓN

Tabla 15: formulario de facturación

Historia de Usuario		
Número: 1	Nombre: formulario de facturación (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :2	
Descripción: Yo, como administrador/empleado, espero poder generar vouchers para establecer la hora de llegada del cliente, el tipo de servicios y demás datos básicos. Además, se desea poder facturar sobre este voucher generado		
Observaciones: La factura no puede modificarse La facturación debe generar ingresos		

Ref: Autor

PRÉSTAMOS

Tabla 16: formulario de préstamo

Historia de Usuario		
Número: 1	Nombre: formulario de préstamo (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :2	
Descripción: Yo, como administrador/empleado, espero poder realizar préstamos del inventario actual disponible y facturarlos, generando un ingreso en el sistema		
Observaciones: La bitácora no puede ser modificada o eliminada El préstamo debe generar ingresos		

Ref: Autor

ESTADÍSTICAS

Tabla 17: estadísticas de ingresos y egresos

Historia de Usuario		
Número: 1	Nombre: Estadísticas de ingresos y egresos (A, E)	
Prioridad en Negocio: 1	Iteración Asignada: :2	
Descripción: Yo como administrador o empleado espero poder observar el estado de ingresos y egresos, en un rango de fechas determinado		
Observaciones: Se deben usar gráficas		

Ref: Autor

Otras Interfaces

Tabla 18: auditoría

Historia De Usuario	
Numero: 1	Nombre: Auditoria (A)
Prioridad en Negocio: 2	Iteración Asignada: 2
Descripción: Yo como administrador deseo poder ver cada transacción de creación, modificación o eliminación sobre el sistema, junto con el usuario que la realizó	
Observaciones: La auditoría no puede ser modificada o eliminada	

Ref: Autor

Requerimientos de software

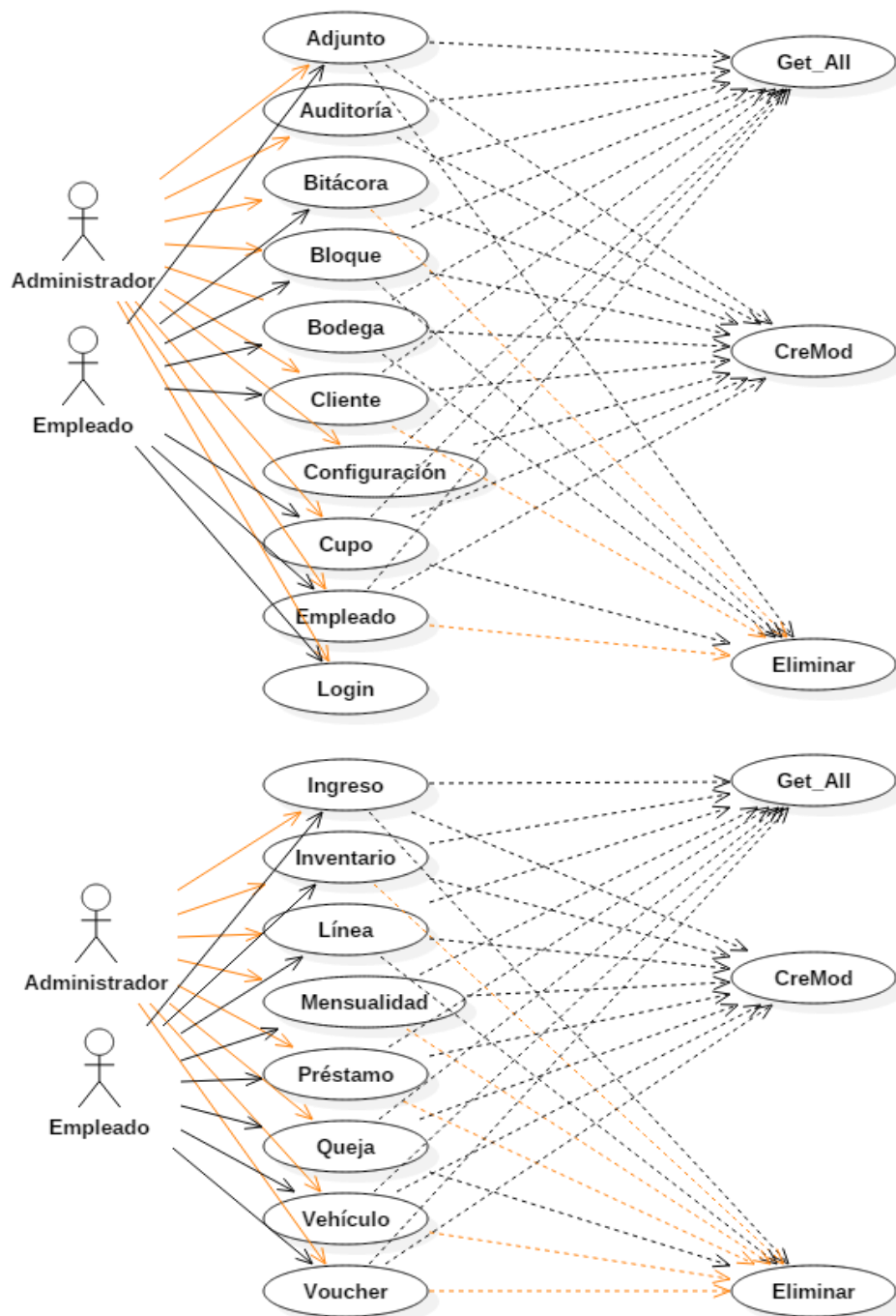
Tabla 19: requerimientos de software

REQUERIMIENTO	DESCRIPCIÓN
R1	Ejecución
R2	Conexión a la base de datos
R3	Notificaciones
R4	Consultar base de datos
R5	Crear registro en la base de datos
R6	Modificar registro en la base de datos
R7	Eliminar registro en la base de datos
R8	Autenticación de usuario

Ref: Autor

Casos de uso

Ilustración 1: diagrama de casos de uso



Ref: Autor

Descripción de casos de uso

TIPOS DE FUNCIÓN

En la anterior ilustración se pueden resaltar tres funciones principales que coinciden con la mayoría de clases; a continuación, se detalla su uso:

- **Get_All_x**

La función Get_All, es la función que se encarga de obtener los datos de la base de datos, y devolverlos en un arreglo de estructura, que tiene el mismo nombre de la clase. Esta función obtiene los datos a partir de consultas dinámicas, construidas a partir de los parámetros enviados.

- **CreMod_x**

La función CreMod, es la función que se encarga de **crear** o **modificar** los datos, en la base de datos, devolviendo un **1** como respuesta de éxito, o un mensaje de error.

- **Eliminar_x**

La función Eliminar, es la función encargada de eliminar (lógicamente, en la mayoría de los casos) los registros de la base de datos.

ADJUNTO

Tabla 20: ver adjuntos

Caso de uso	Ver adjuntos
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar un documento adjunto en el sistema, desde el listado de adjuntos
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un documento en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver un documento en el sistema. El usuario accede al módulo adjuntos .	2. El sistema despliega el listado de adjuntos creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de adjuntos, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	

Acción 5: si el documento no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente

Ref: Autor

Tabla 21: crear adjunto

Caso de uso	Crear adjunto
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un documento adjunto en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un documento en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar un documento en el sistema. El usuario accede al módulo adjuntos .	2. El sistema despliega el listado de adjuntos creados actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de adjuntos
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 22: modificar adjunto

Caso de uso	Modificar adjunto
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar un documento adjunto en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un documento en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar un documento en el sistema. El usuario accede al módulo adjuntos .	2. El sistema despliega el listado de adjuntos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el documento a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 23: eliminar adjunto

Caso de uso	Eliminar adjunto
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda eliminar un documento adjunto en el sistema

Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un documento del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide eliminar un documento en el sistema. El usuario accede al módulo adjuntos .	2. El sistema despliega el listado de adjuntos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el documento a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el documento a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de documentos adjuntos.
Curso Alterno	
Acción 6: si el documento a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del documento no afecte la integridad de la base de datos	

Ref: Autor

AUDITORIA

Tabla 24: ver auditoría

Caso de uso	Ver auditoría
Actores	Administrador
Propósito	Este caso de uso permite que se pueda consultar la auditoría en el sistema, desde el listado de registros de auditoría
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un registro de auditoría en el sistema
Tipo	esencial

Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver un registro de auditoría en el sistema. El usuario accede al módulo auditoría .	2. El sistema despliega el listado de auditoría creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de registros de auditoría, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alternativo	
Acción 5: si el registro no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

BITÁCORA

Tabla 25: ver bitácora

Caso de uso	Ver bitácora
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar la bitácora en el sistema, desde el listado de adjuntos
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un mensaje de bitácora en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver un mensaje de bitácora en el sistema. El usuario accede al módulo bitácora .	2. El sistema despliega el listado de mensajes de bitácora creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de mensajes, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alternativo	

Acción 5: si el mensaje no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente

Ref: Autor

Tabla 26: crear bitácora

Caso de uso	Crear mensaje de bitácora
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un mensaje de bitácora en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un mensaje de bitácora en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar un mensaje en el sistema. El usuario accede al módulo bitácora .	2. El sistema despliega el listado de mensajes creados actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de mensaje de bitácora
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 27: modificar bitácora

Caso de uso	Modificar mensaje de bitácora
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar un documento adjunto en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un mensaje de bitácora en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar un mensaje de bitácora en el sistema. El usuario accede al módulo bitácora .	2. El sistema despliega el listado de mensajes creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el documento a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 28: eliminar bitácora

Caso de uso	Eliminar bitácora
Actores	Administrador
Propósito	Este caso de uso permite que se pueda eliminar un mensaje de bitácora en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un mensaje de bitácora del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando el usuario administrador decide eliminar un documento en el sistema. El usuario accede al módulo bitácora .	2. El sistema despliega el listado de mensajes de bitácora creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el mensaje a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el mensaje a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de mensjaes de bitácora.
Curso Alterno	
Acción 6: si el mensaje a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del documento no afecte la integridad de la base de datos	

Ref: Autor

BLOQUE

Tabla 29: ver bloques

Caso de uso	Ver bloques
Actores	Administrador, Empleado

Propósito	Este caso de uso permite que se pueda consultar la información de un bloque en el sistema, desde el listado de bloques
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un bloque en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver la información de un bloque en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de bloques creados actualmente, junto con sus líneas y cupos, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de bloques, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alternativo	
Acción 5: si el bloque no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 30: crear bloques

Caso de uso	Crear bloque
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un bloque en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un bloque en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar un bloque en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de bloques creados actualmente en el sistema, junto con sus líneas y cupos, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de bloques.

5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 31: modificar bloque

Caso de uso	Modificar bloque
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar la información de un bloque en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un bloque en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar la información de un bloque en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de bloques creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el bloque a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados

7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 32: eliminar bloque

Caso de uso	Eliminar bloque
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda eliminar un bloque en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un bloque del sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide eliminar un bloque en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de bloques creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el bloque a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el bloque a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de bloques.
Curso Alterno	
Acción 6: si el bloque a eliminar no puede ser eliminado, se muestra un mensaje de error	

Sección Garantizar Seguridad.

1. El sistema verifica que la eliminación del bloque no afecte la integridad de la base de datos

Ref: Autor

BODEGA

Tabla 33: ver bodegas

Caso de uso	Ver bodegas
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar una bodega en el sistema, desde el listado de bodegas
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar una bodega en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide buscar o ver una bodega en el sistema. El usuario accede al módulo bodegas .	2. El sistema despliega el listado de bodegas creadas actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de bodegas, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	
Acción 5: si el documento no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 34: crear bodega

Caso de uso	Crear bodega
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear una bodega en el sistema

Resumen	Este caso de uso ocurre cuando el usuario quiere registrar una bodega en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar una bodega en el sistema. El usuario accede al módulo bodegas .	2. El sistema despliega el listado de bodegas creadas actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de bodega
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: autor

Tabla 35: modificar bodega

Caso de uso	Modificar bodega
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar una bodega en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar una bodega en el sistema
Tipo	No esencial
Curso Normal de los Eventos	

Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar una bodega en el sistema. El usuario accede al módulo bodegas .	2. El sistema despliega el listado de bodegas creadas actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra la bodega a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 36: eliminar bodega

Caso de uso	Eliminar bodega
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda eliminar una bodega en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar una bodega del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide eliminar una bodega en el sistema. El usuario accede al módulo bodegas .	2. El sistema despliega el listado de bodegas creadas actualmente en el sistema, dentro de una tabla.

3. El usuario selecciona o filtra la bodega a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que la bodega a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza el listado de bodegas.
Curso Alterno	
Acción 6: si la bodega a eliminar no puede ser eliminada, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación de la bodega no afecte la integridad de la base de datos	

Ref: Autor

CLIENTE

Tabla 37: ver cliente

Caso de uso	Ver clientes
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar un cliente en el sistema, desde el listado de clientes
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un cliente en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide buscar o ver un cliente en el sistema. El usuario accede al módulo clientes .	2. El sistema despliega el listado de clientes creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de clientes, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	

Acción 5: si el cliente no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente

Ref: Autor

Tabla 38: crear cliente

Caso de uso	Crear cliente
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un cliente en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un cliente en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar un cliente en el sistema. El usuario accede al módulo clientes .	2. El sistema despliega el listado de clientes creados actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de clientes
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 39: modificar cliente

Caso de uso	Modificar cliente
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar un cliente en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un cliente en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide modificar un cliente en el sistema. El usuario accede al módulo clientes .	2. El sistema despliega el listado de clientes creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el documento a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 40: eliminar cliente

Caso de uso	Eliminar cliente
Actores	Administrador

Propósito	Este caso de uso permite que se pueda eliminar un cliente en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un cliente del sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide eliminar un cliente en el sistema. El usuario accede al módulo clientes .	2. El sistema despliega el listado de clientes creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el cliente a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el cliente a eliminar, pueda ser eliminado (integridad referencial), mensualidades, vehículos y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza el listado de clientes.
Curso Alterno	
Acción 6: si el cliente a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del documento no afecte la integridad de la base de datos 2. Un cliente sólo puede ser eliminado por el usuario administrador	

Ref: Autor

CONFIGURACIÓN

Tabla 41: ver configuración

Caso de uso	Ver configuración
Actores	Administrador
Propósito	Este caso de uso permite que se pueda consultar la configuración en el sistema.
Resumen	Este caso de uso ocurre cuando el usuario quiere ver la configuración del sistema.
Tipo	Esencial

Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide ver la configuración del sistema. El usuario accede al módulo configuración .	2. El sistema despliega la configuración actual del sistema
Curso Alterno	
Acción 5: si no existe una configuración actual, el sistema muestra los campos en blanco	

Ref: Autor

Tabla 42: crear configuración

Caso de uso	Crear/Modificar configuración
Actores	Administrador
Propósito	Este caso de uso permite que se pueda establecer la configuración sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere establecer la configuración del sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide ver la configuración del sistema. El usuario accede al módulo configuración .	2. El sistema despliega la configuración actual del sistema
3. El usuario modifica los campos necesarios y da clic en guardar	4. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	

Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5

Sección Garantizar Seguridad.

1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente

Ref: Autor

CUPO

Tabla 43: ver cupo

Caso de uso	Ver cupos
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar un cupo en el sistema, desde el listado de cupos
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un cupo en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide buscar o ver un cupo en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de cupos creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de cupos, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	
Acción 5: si el cupo no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 44: crear cupo

Caso de uso	Crear cupo
Actores	Administrador, Empleado

Propósito	Este caso de uso permite que se pueda crear un cupo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un cupo en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide registrar un cupo en el sistema. El usuario accede al módulo adjuntos .	2. El sistema despliega el listado de cupos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona un bloque, línea, y da clic en nuevo .	4. El sistema muestra el formulario de cupos
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alternativo	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 45: modificar cupo

Caso de uso	Modificar cupo
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar un cupo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un cupo en el sistema
Tipo	Esencial

Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide modificar un cupo en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de cupos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el cupo, del bloque y línea a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 46: eliminar cupo

Caso de uso	Eliminar cupo
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda eliminar un cupo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un cupo del sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide eliminar un cupo en el sistema.	2. El sistema despliega el listado de cupos creados actualmente en el sistema, dentro de una tabla.

El usuario accede al módulo cupos .	
3. El usuario selecciona o filtra el cupo a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el cupo a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza el listado de cupos.
Curso Alterno	
Acción 6: si el cupo a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del cupo no afecte la integridad de la base de datos	

Ref: Autor

EMPLEADO

Tabla 47: ver empleados

Caso de uso	Ver empleados
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar un empleado en el sistema, desde el listado de empleados
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un empleado en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide buscar o ver un empleado en el sistema. El usuario accede al módulo empleados .	2. El sistema despliega el listado de empleados creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de empleados, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.

Curso Alterno	
Acción 5: si el empleado no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 48: crear empleado

Caso de uso	Crear empleado
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un empleado en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un empleado en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar un empleado en el sistema. El usuario accede al módulo empleados .	2. El sistema despliega el listado de empleados creados actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de empleados
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 49: modificar empleado

Caso de uso	Modificar empleado
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar un empleado en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un empleado en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar un empleado en el sistema. El usuario accede al módulo empleados .	2. El sistema despliega el listado de empleados creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el empleado a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 50: eliminar empleado

Caso de uso	Eliminar empleado
Actores	Administrador

Propósito	Este caso de uso permite que se pueda eliminar un empleado en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un empleado del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide eliminar un empleado en el sistema. El usuario accede al módulo adjuntos .	2. El sistema despliega el listado de adjuntos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el documento a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el empleado a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de empleados.
Curso Alterno	
Acción 6: si el empleado a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del empleado no afecte la integridad de la base de datos 2. La eliminación sólo puede ser realizada por el usuario administrador	

Ref: Autor

INGRESO

Tabla 51: ver ingreso

Caso de uso	Ver ingreso
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar un ingreso o egreso en el sistema, desde el listado de ingresos y egresos

Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un ingreso o egreso en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide buscar o ver un ingreso o egreso en el sistema. El usuario accede al módulo ingresos .	2. El sistema despliega el listado de ingresos y egresos creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de ingresos y egresos, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	
Acción 5: si el ingreso o egreso no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 52: crear ingreso o egreso

Caso de uso	Crear ingreso o egreso
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un ingreso o egreso en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un ingreso o egreso en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide registrar un ingreso o egreso en el sistema. El usuario accede al módulo ingresos .	2. El sistema despliega el listado de ingresos y egresos creados actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de ingreso o egreso

5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 53: eliminar ingreso o egreso

Caso de uso	Eliminar ingreso o egreso
Actores	Administrador
Propósito	Este caso de uso permite que se pueda eliminar un ingreso o egreso en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un ingreso o egreso del sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando un usuario decide eliminar un ingreso o egreso en el sistema. El usuario accede al módulo ingresos y egresos .	2. El sistema despliega el listado de ingresos y/o egresos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el registro a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el ingreso o egreso a eliminar, pueda ser eliminado (integridad referencial) y lo elimina

7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de ingresos y egresos.
Curso Alterno	
Acción 6: si el ingreso o egreso a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del ingreso o egreso no afecte la integridad de la base de datos	

INVENTARIO

Tabla 54: ver inventario

Caso de uso	Ver inventario
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar el inventario en el sistema, desde el listado de inventario
Resumen	Este caso de uso ocurre cuando el usuario quiere ver el inventario en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver un ítem de inventario en el sistema. El usuario accede al módulo inventario .	2. El sistema despliega el listado de inventario creado actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de inventario, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	
Acción 5: si el inventario no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 55: crear inventario

Caso de uso	Crear inventario
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un ítem de inventario en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un ítem de inventario en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar un ítem de inventario en el sistema. El usuario accede al módulo inventario .	2. El sistema despliega el listado de inventario creado actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de inventario
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 56: modificar inventario

Caso de uso	Modificar inventario
Actores	Administrador, Empleado

Propósito	Este caso de uso permite que se pueda modificar un ítem de inventario en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un ítem de inventario en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar un ítem de inventario en el sistema. El usuario accede al módulo inventario .	2. El sistema despliega el listado de ítems de inventario creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el ítem a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 57: eliminar inventario

Caso de uso	Eliminar inventario
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda eliminar un ítem de inventario en el sistema

Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un ítem de inventario del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide eliminar un ítem de inventario en el sistema. El usuario accede al módulo adjuntos .	2. El sistema despliega el listado de ítems creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el ítem a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el documento a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de inventario.
Curso Alterno	
Acción 6: si el ítem a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del documento no afecte la integridad de la base de datos	

Ref: Autor

LÍNEA

Tabla 58: ver línea

Caso de uso	Ver líneas
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar una línea en el sistema, desde el listado de adjuntos
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar una línea en el sistema
Tipo	Esencial
Curso Normal de los Eventos	

Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver una línea en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de líneas creadas actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de líneas, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	
Acción 5: si la línea no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 59: crear línea

Caso de uso	Crear línea
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear una línea en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar una línea en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar una línea en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de líneas creadas actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de línea
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	

Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5

Sección Garantizar Seguridad.

1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente

Ref: Autor

Tabla 60: modificar línea

Caso de uso	Modificar línea
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar una línea en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar una línea en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar una línea en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de líneas creadas actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra la línea a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	

1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente

Ref: Autor

Tabla 61: eliminar línea

Caso de uso	Eliminar línea
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda eliminar una línea en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar una línea del sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide eliminar una línea en el sistema. El usuario accede al módulo bloques .	2. El sistema despliega el listado de líneas creadas actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra la línea a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que la línea a eliminar, pueda ser eliminada (integridad referencial) y la elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de líneas.
Curso Alternativo	
Acción 6: si el documento a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del documento no afecte la integridad de la base de datos	

Ref: Autor

MENSUALIDAD

Tabla 62: ver mensualidad

Caso de uso	Ver mensualidad
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar una mensualidad en el sistema, desde el listado de mensualidades
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar una mensualidad en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver una mensualidad en el sistema. El usuario accede al módulo mensualidades .	2. El sistema despliega el listado de adjuntos creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de adjuntos, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	
Acción 5: si la mensualidad no aparece en la tabla, debe verificarse que esté creada, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 63: crear mensualidad

Caso de uso	Crear mensualidad
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear una mensualidad en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar una mensualidad en el sistema
Tipo	Esencial
Curso Normal de los Eventos	

Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar una mensualidad en el sistema. El usuario accede al módulo mensualidades .	2. El sistema despliega el listado de mensualidades creadas actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de mensualidades
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 64: modificar mensualidad

Caso de uso	Modificar mensualidad
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar una mensualidad en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar una mensualidad en el sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema

1. Este caso de uso inicia cuando una persona decide modificar una mensualidad en el sistema. El usuario accede al módulo mensualidades .	2. El sistema despliega el listado de mensualidades creadas actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el documento a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 65: eliminar mensualidad

Caso de uso	Eliminar mensualidad
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda eliminar una mensualidad en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar una mensualidad del sistema
Tipo	Esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide eliminar una mensualidad en el sistema. El usuario accede al módulo mensualidades .	2. El sistema despliega el listado de mensualidades creadas actualmente en el sistema, dentro de una tabla.

3. El usuario selecciona o filtra la mensualidad a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el documento a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de mensualidades.
Curso Alterno	
Acción 6: si la mensualidad a eliminar no puede ser eliminada, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del documento no afecte la integridad de la base de datos	

Ref: Autor

PRÉSTAMO

Tabla 66: ver préstamo

Caso de uso	Ver préstamos
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar un préstamo en el sistema, desde el listado de préstamos
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un préstamo en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver un préstamo en el sistema. El usuario accede al módulo préstamos .	2. El sistema despliega el listado de préstamos creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de préstamos, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.

Curso Alterno	
Acción 5: si el préstamo no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 67: crear préstamo

Caso de uso	Crear préstamo
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un préstamo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un préstamo en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar un préstamo en el sistema. El usuario accede al módulo préstamos .	2. El sistema despliega el listado de préstamos creados actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de préstamos
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 68: modificar préstamo

Caso de uso	Modificar préstamo
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar un préstamo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un préstamo en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar un préstamo en el sistema. El usuario accede al módulo préstamos .	2. El sistema despliega el listado de préstamos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el préstamo a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 69: eliminar préstamo

Caso de uso	Eliminar préstamo
Actores	Administrador, Empleado

Propósito	Este caso de uso permite que se pueda eliminar un préstamo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un préstamo del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide eliminar un préstamo en el sistema. El usuario accede al módulo préstamos .	2. El sistema despliega el listado de préstamos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el préstamo a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el préstamo a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de préstamos.
Curso Alterno	
Acción 6: si el préstamo a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del documento no afecte la integridad de la base de datos	

Ref: Autor

QUEJA

Tabla 70: ver queja

Caso de uso	Ver queja
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar una queja en el sistema, desde el listado de quejas
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar una queja en el sistema

Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver una queja en el sistema. El usuario accede al módulo quejas .	2. El sistema despliega el listado de quejas creadas actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de quejas, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	
Acción 5: si la queja no aparece en la tabla, debe verificarse que esté creada, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 71: crear queja

Caso de uso	Crear queja
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear una queja en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar una queja en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar una queja en el sistema. El usuario accede al módulo quejas .	2. El sistema despliega el listado de quejas creadas actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de quejas
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	

Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 72: modificar queja

Caso de uso	Modificar queja
Actores	Administrador
Propósito	Este caso de uso permite que se pueda modificar una queja en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar una queja en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar una queja en el sistema. El usuario accede al módulo quejas .	2. El sistema despliega el listado de quejas creadas actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra la queja a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	

1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente

Ref: Autor

Tabla 73: eliminar queja

Caso de uso	Eliminar queja
Actores	Administrador
Propósito	Este caso de uso permite que se pueda eliminar una queja en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar una queja del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide eliminar una queja. El usuario accede al módulo quejas .	2. El sistema despliega el listado de quejas creadas actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra la queja a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que la queja a eliminar, pueda ser eliminado (integridad referencial) y la elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de quejas.
Curso Alterno	
Acción 6: si el documento a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación de la queja no afecte la integridad de la base de datos	

Ref: Autor

VEHÍCULO

Tabla 74: ver vehículo

Caso de uso	Ver vehículo
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar un vehículo en el sistema, desde el listado de adjuntos
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un vehículo en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver un vehículo en el sistema. El usuario accede al módulo vehículos .	2. El sistema despliega el listado de vehículos creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de vehículos, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alternativo	
Acción 5: si el vehículo no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 75: crear vehículo

Caso de uso	Crear vehículo
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un vehículo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un vehículo en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema

1. Este caso de uso inicia cuando una persona decide registrar un vehículo en el sistema. El usuario accede al módulo vehículos .	2. El sistema despliega el listado de vehículos creados actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de vehículos
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 76: modificar vehículo

Caso de uso	Modificar vehículo
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda modificar un vehículo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un vehículo en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar un vehículo en el sistema. El usuario accede al módulo adjuntos .	2. El sistema despliega el listado de vehículos creados actualmente en el sistema, dentro de una tabla.

3. El usuario selecciona o filtra el vehículo a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 77: eliminar vehículo

Caso de uso	Eliminar vehículo
Actores	Administrador
Propósito	Este caso de uso permite que se pueda eliminar un vehículo en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un vehículo del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide eliminar un vehículo en el sistema. El usuario accede al módulo vehículos .	2. El sistema despliega el listado de vehículos creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el vehículo a eliminar	4. El sistema habilita los botones modificar y eliminar

5. El usuario da clic en eliminar .	6. El sistema verifica que el vehículo a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de vehículos.
Curso Alterno	
Acción 6: si el vehículo a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del vehículo no afecte la integridad de la base de datos	

Ref: Autor

VOUCHER

Tabla 78: ver voucher

Caso de uso	Ver voucher
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda consultar un voucher en el sistema, desde el listado de vouchers
Resumen	Este caso de uso ocurre cuando el usuario quiere ver o buscar un voucher en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide buscar o ver un voucher en el sistema. El usuario accede al módulo voucher .	2. El sistema despliega el listado de vouchers creados actualmente en el sistema, dentro de una tabla, junto con sus datos.
3. El usuario puede observar el listado de vouchers, o filtrar desde los campos de la parte superior	4. El sistema utiliza el filtro, para mostrar los resultados dentro de la misma tabla, cada vez que se digita sobre un campo filtro.
Curso Alterno	
Acción 5: si el voucher no aparece en la tabla, debe verificarse que esté creado, o que se hayan diligenciado los filtros correctamente	

Ref: Autor

Tabla 79: crear voucher

Caso de uso	Crear voucher
Actores	Administrador, Empleado
Propósito	Este caso de uso permite que se pueda crear un voucher en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere registrar un voucher en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide registrar un voucher en el sistema. El usuario accede al módulo vouchers .	2. El sistema despliega el listado de vouchers creados actualmente en el sistema, dentro de una tabla.
3. El usuario da clic en nuevo .	4. El sistema muestra el formulario de vouchers
5. El usuario diligencia el formulario	6. El sistema verifica que se hayan diligenciado los datos obligatorios mostrando un mensaje de error (si lo hay). Cuando los datos sean correctos, el sistema guarda los datos en la base de datos y muestra un mensaje de éxito
7. El usuario recibe el mensaje de éxito	
Curso Alterno	
Acción 6: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 80: modificar voucher

Caso de uso	Modificar voucher
Actores	Administrador
Propósito	Este caso de uso permite que se pueda modificar un voucher en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere modificar un voucher en el sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide modificar un voucher en el sistema. El usuario accede al módulo vouchers .	2. El sistema despliega el listado de vouchers creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el voucher a modificar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en modificar .	6. El sistema abre el formulario correspondiente, junto con los datos actuales, para ser modificados
7. El usuario diligencia el formulario, cambiando los datos necesarios, y da clic en guardar .	8. El sistema verifica que los datos se hayan diligenciado correctamente.
9. El usuario recibe el mensaje de guardado.	
Curso Alterno	
Acción 8: Si el usuario ingresa datos inconsistentes, datos incompletos o no a llenado campos obligatorios, el sistema muestra un mensaje solicitando los datos erróneos, faltantes y obligatorios, y regresa al paso 5	
Sección Garantizar Seguridad.	
1. El sistema verifica que cada uno de los datos suministrados por el usuario sean consistentes y estén diligenciados correctamente	

Ref: Autor

Tabla 81: eliminar voucher

Caso de uso	Eliminar voucher
Actores	Administrador
Propósito	Este caso de uso permite que se pueda eliminar un voucher en el sistema
Resumen	Este caso de uso ocurre cuando el usuario quiere eliminar un voucher del sistema
Tipo	No esencial
Curso Normal de los Eventos	
Acción de los actores	Respuesta del sistema
1. Este caso de uso inicia cuando una persona decide eliminar un voucher en el sistema. El usuario accede al módulo vouchers .	2. El sistema despliega el listado de vouchers creados actualmente en el sistema, dentro de una tabla.
3. El usuario selecciona o filtra el voucher a eliminar	4. El sistema habilita los botones modificar y eliminar
5. El usuario da clic en eliminar .	6. El sistema verifica que el voucher a eliminar, pueda ser eliminado (integridad referencial) y lo elimina
7. El usuario recibe un mensaje de éxito.	8. El sistema actualiza la tabla de vouchers.
Curso Alterno	
Acción 6: si el voucher a eliminar no puede ser eliminado, se muestra un mensaje de error	
Sección Garantizar Seguridad.	
1. El sistema verifica que la eliminación del documento no afecte la integridad de la base de datos	

Ref: Autor

Diagramas de secuencia

Creación

Ilustración 2: crear configuración

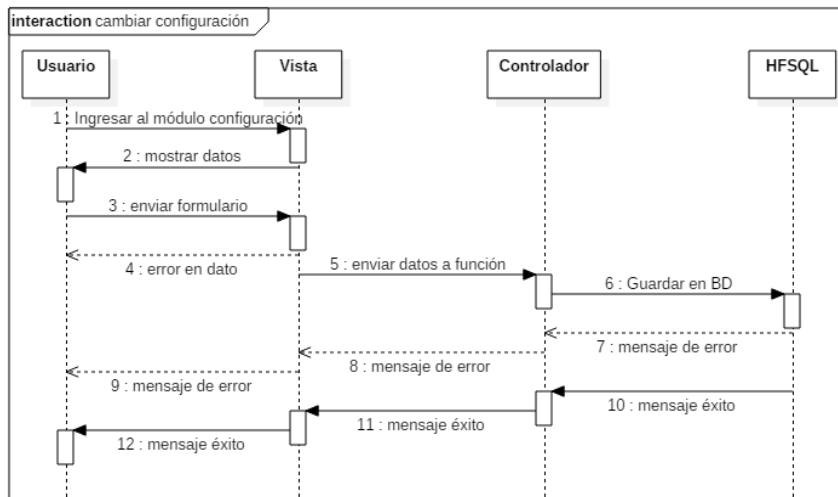
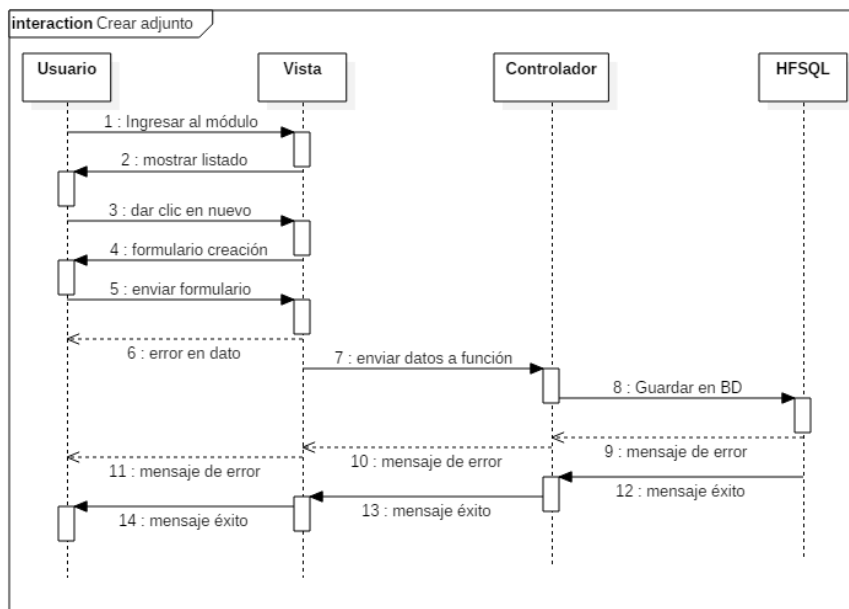
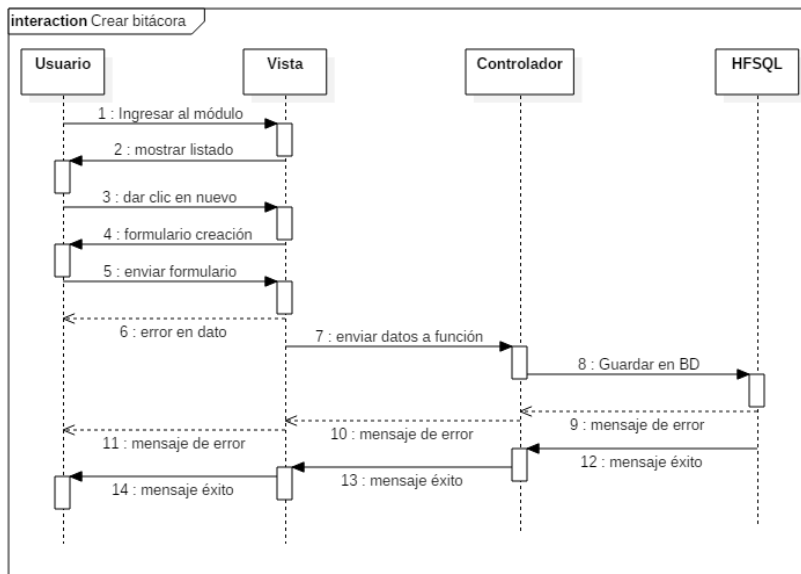


Ilustración 3: crear adjunto



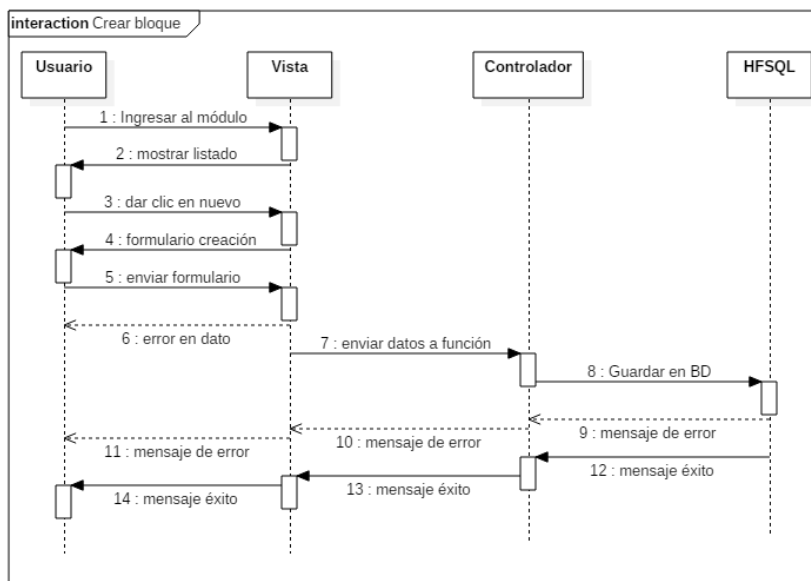
Ref: Autor

Ilustración 4: crear bitácora



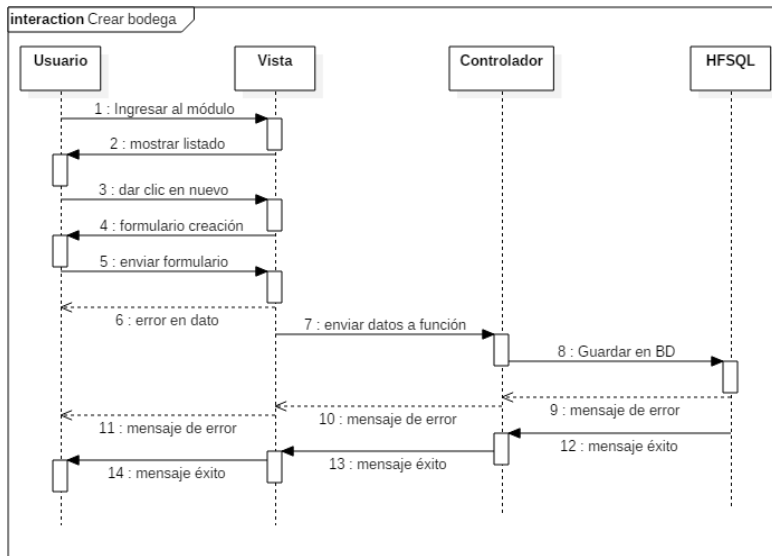
Ref: Autor

Ilustración 5: crear bloque



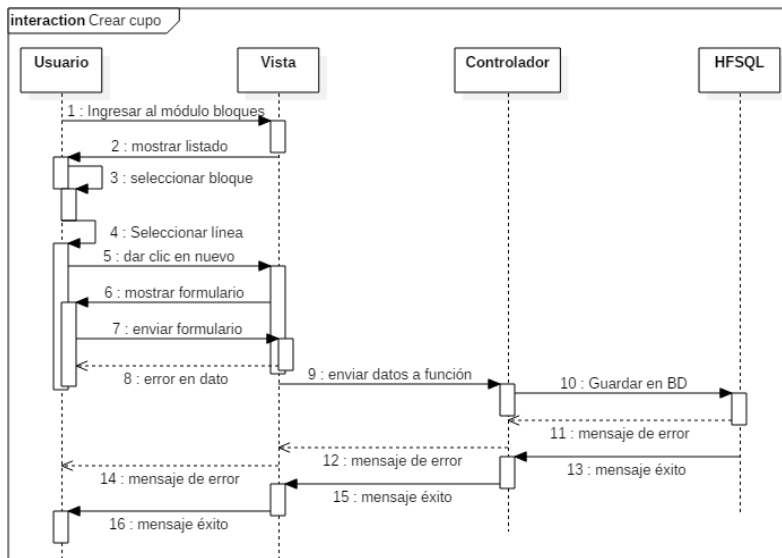
Ref: Autor

Ilustración 6: crear bodega



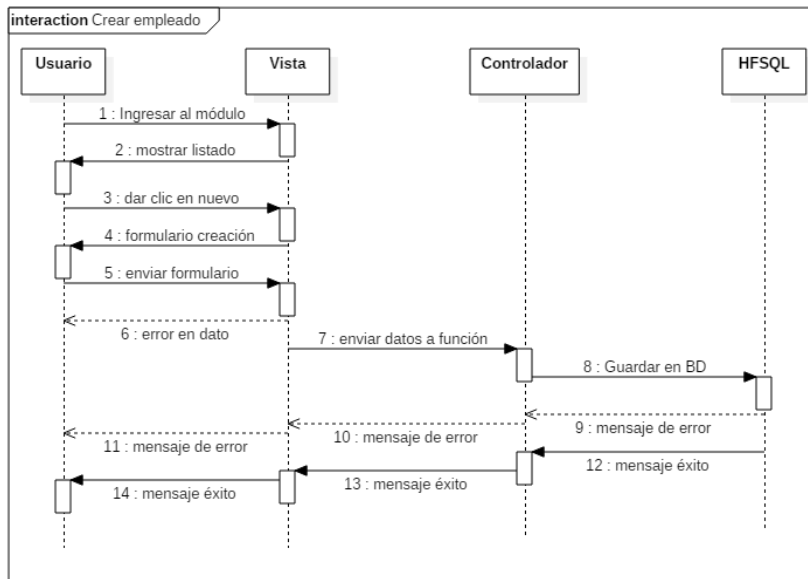
Ref: Autor

Ilustración 7: crear cupo



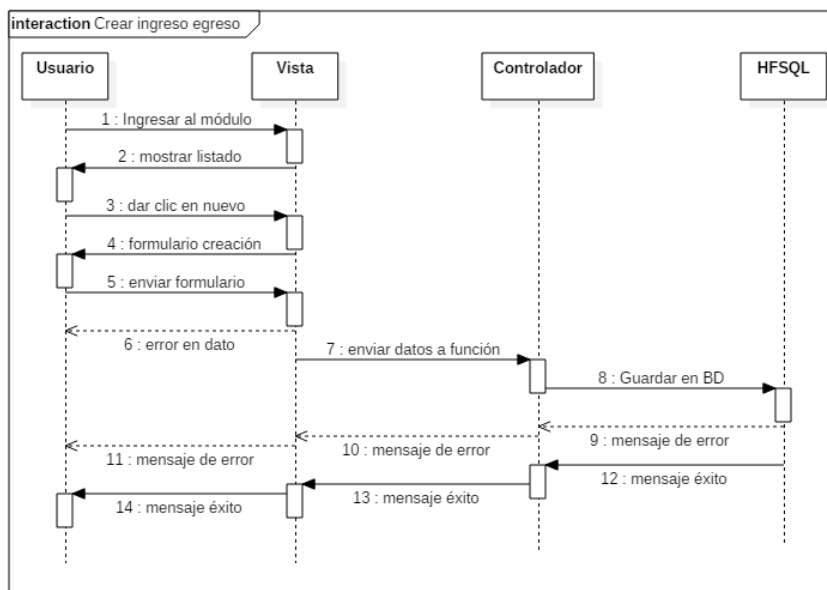
Ref: Autor

Ilustración 8: crear empleado



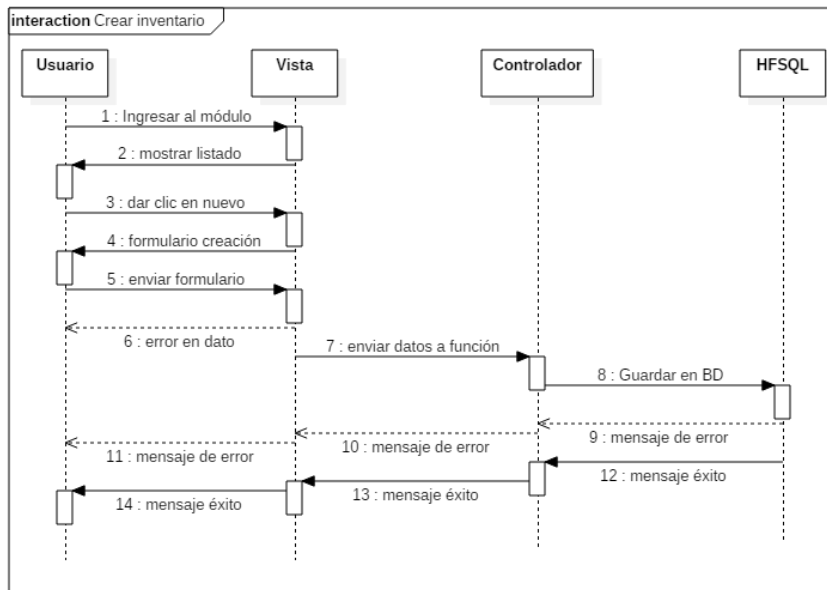
Ref: Autor

Ilustración 9: crear ingreso



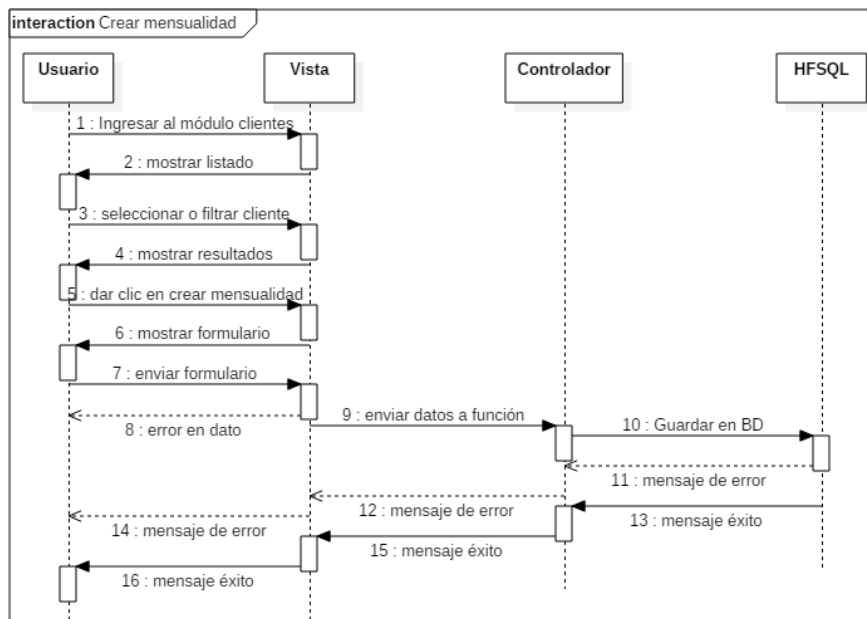
Ref: Autor

Ilustración 10: crear inventario



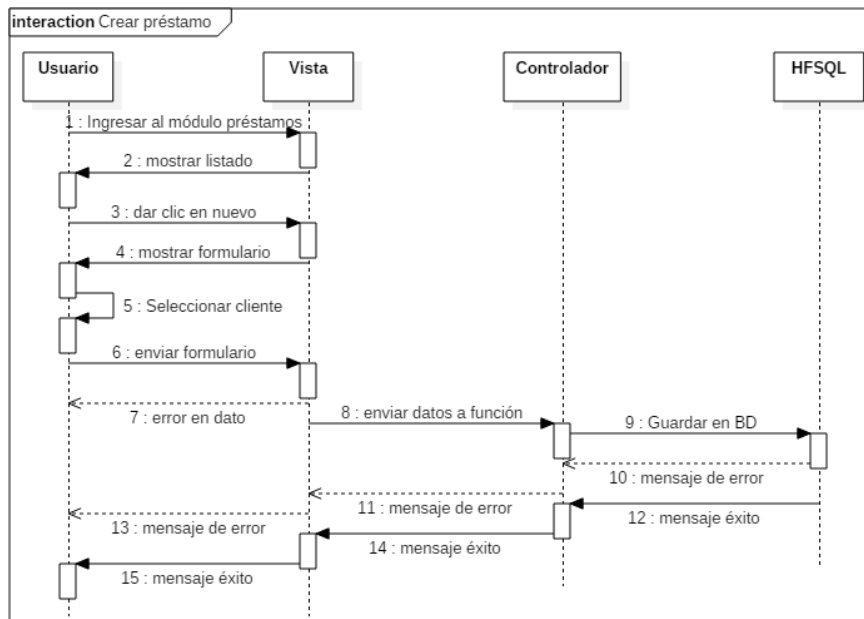
Ref: Autor

Ilustración 11: crear mensualidad



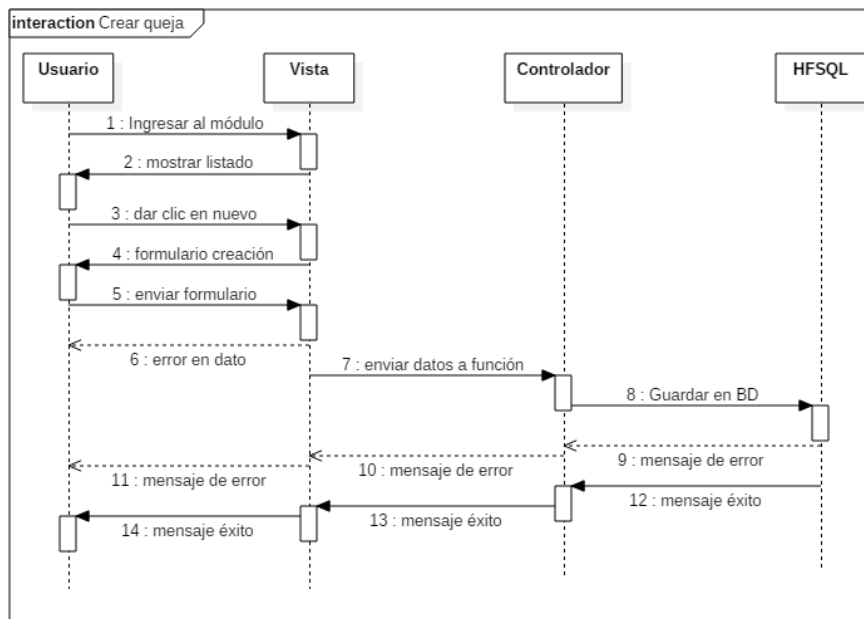
Ref: Autor

Ilustración 12: crear préstamo



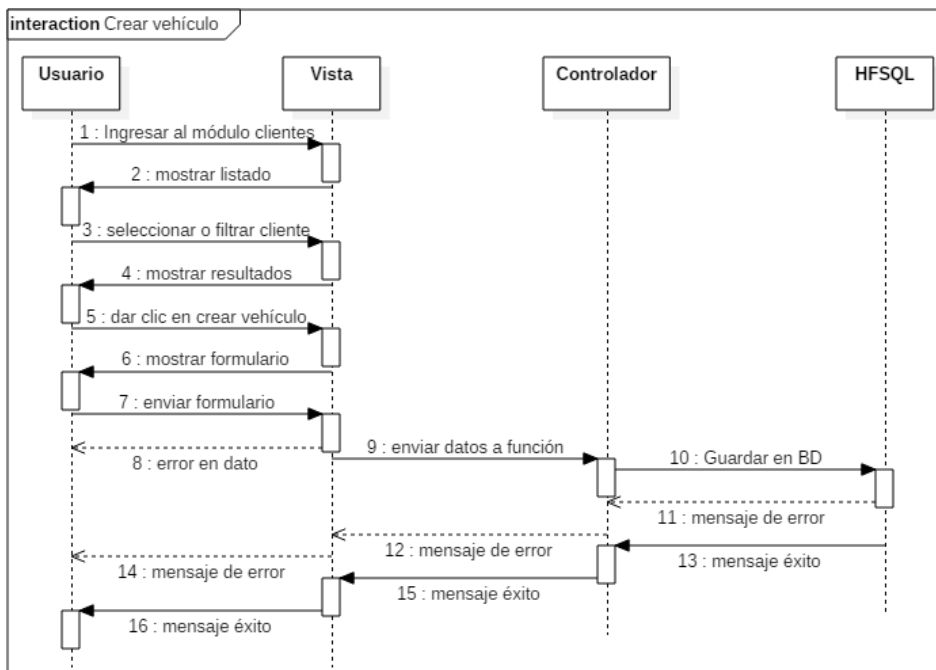
Ref: Autor

Ilustración 13: crear queja



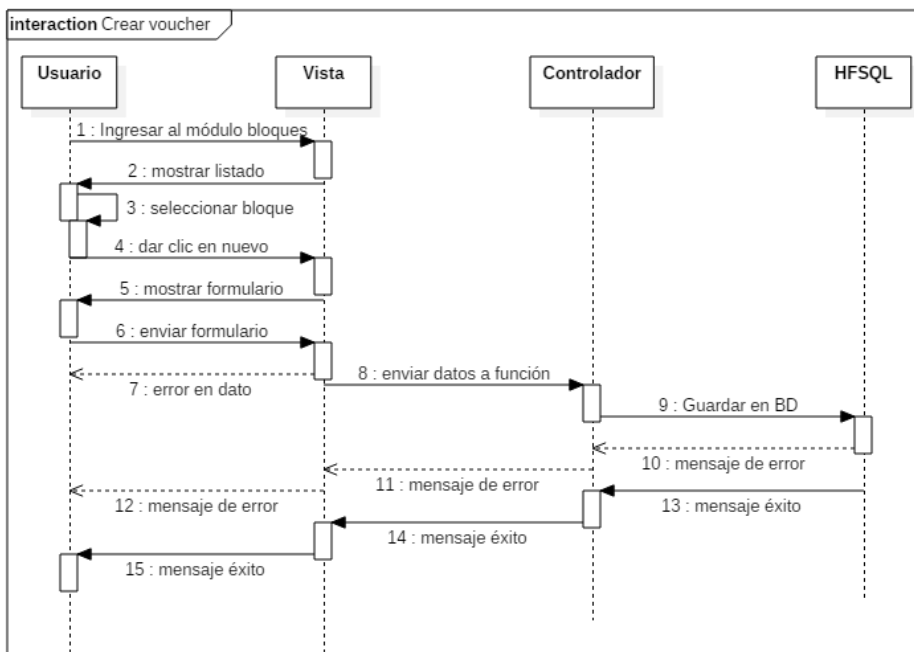
Ref: Autor

Ilustración 14: crear vehículo



Ref: Autor

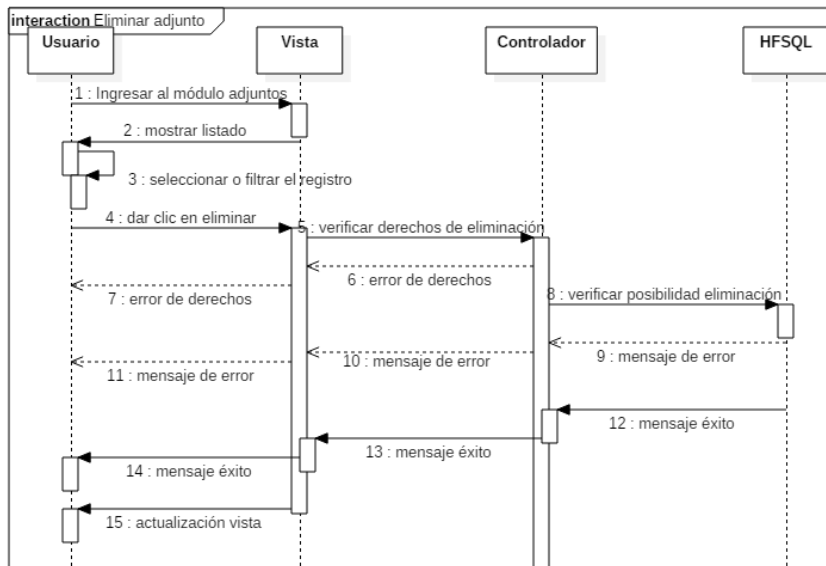
Ilustración 15: crear voucher



Ref: Autor

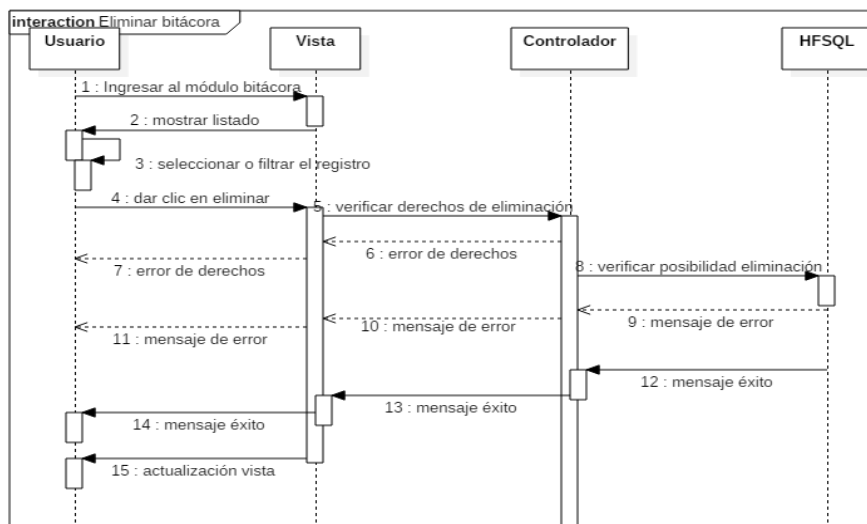
Eliminar

Ilustración 16: eliminar adjunto



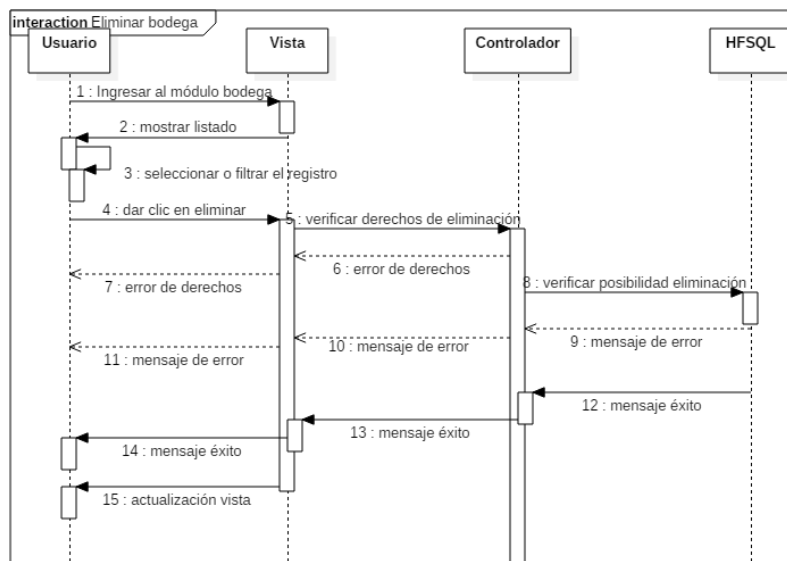
Ref: Autor

Ilustración 17: eliminar bitácora



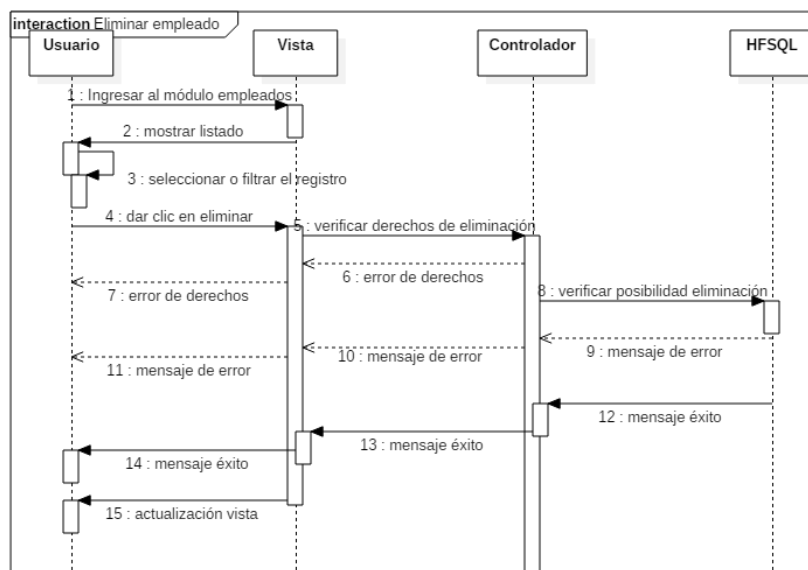
Ref: Autor

Ilustración 18: eliminar bodega



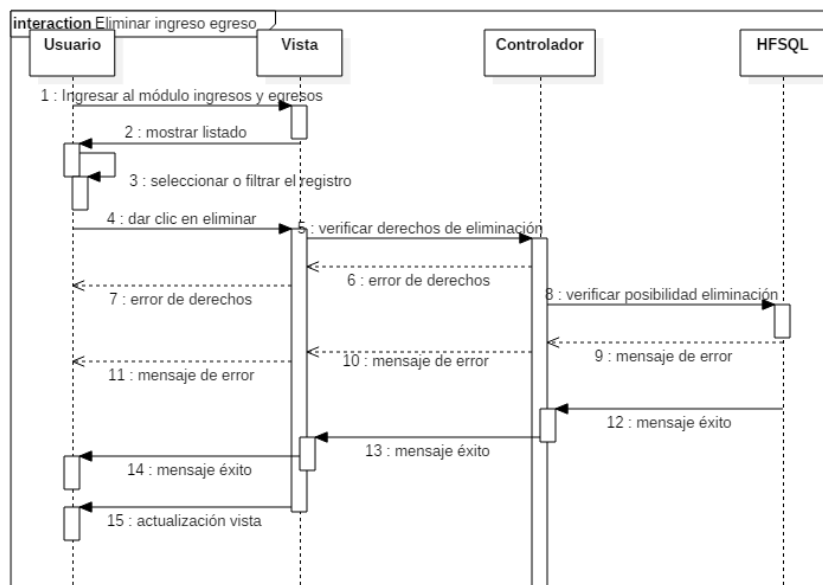
Ref: Autor

Ilustración 19: eliminar empleado



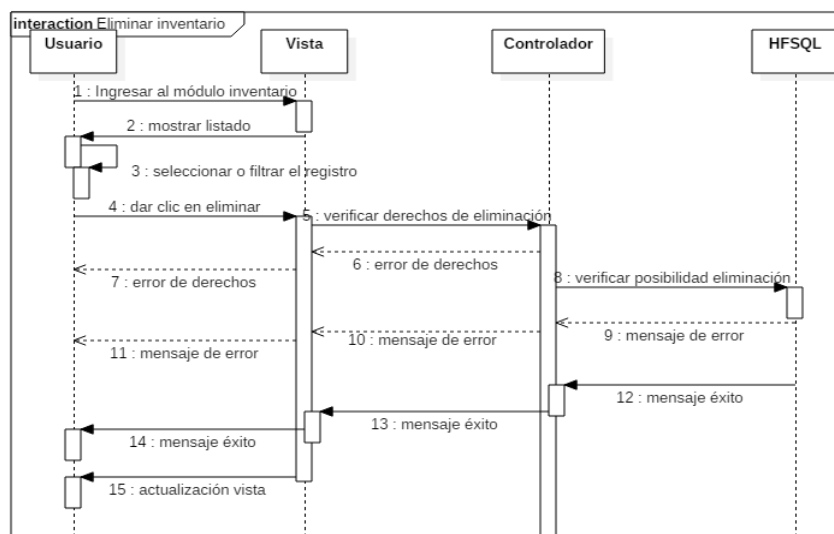
Ref: Autor

Ilustración 20: eliminar ingreso



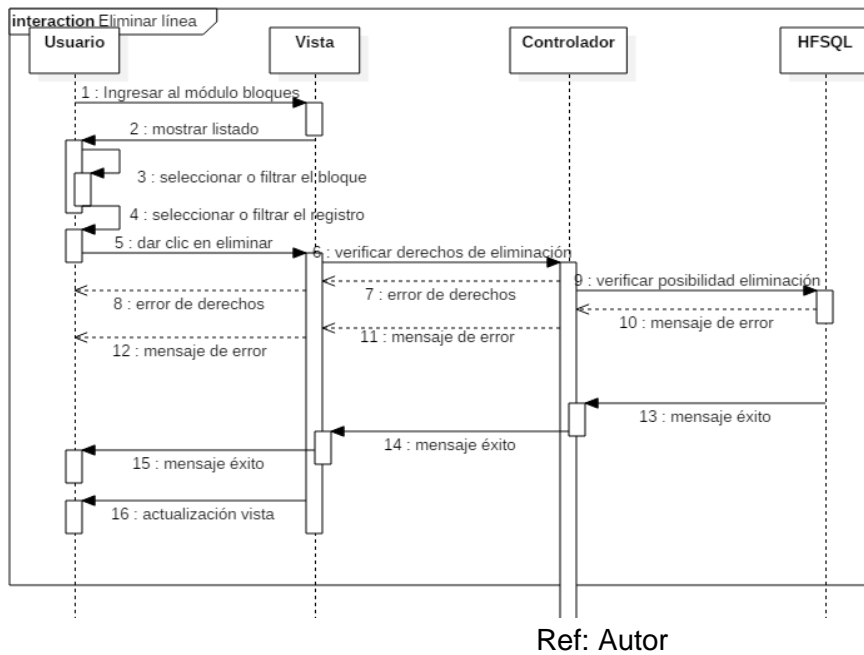
Ref: Autor

Ilustración 21: eliminar inventario



Ref: Autor

Ilustración 22: eliminar línea



Mensualidad

Ilustración 23: eliminar mensualidad

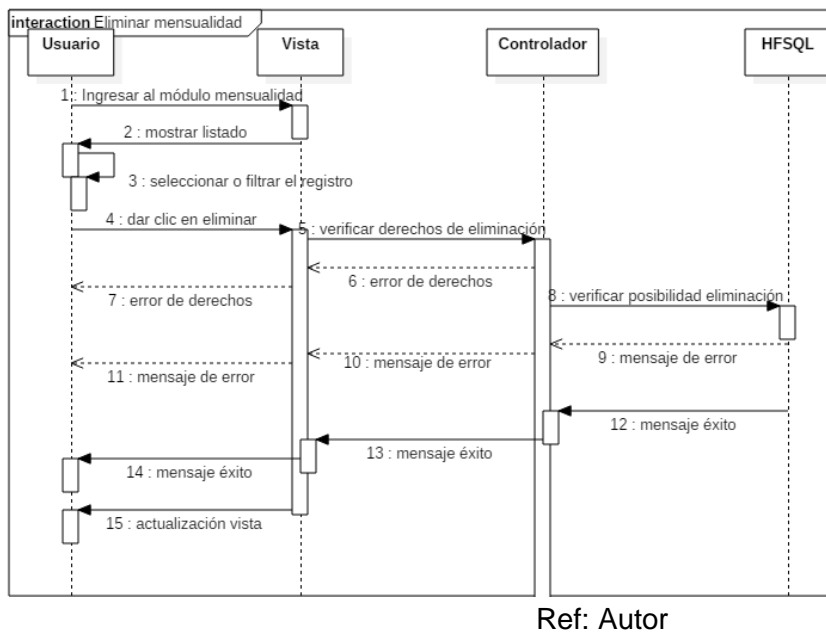
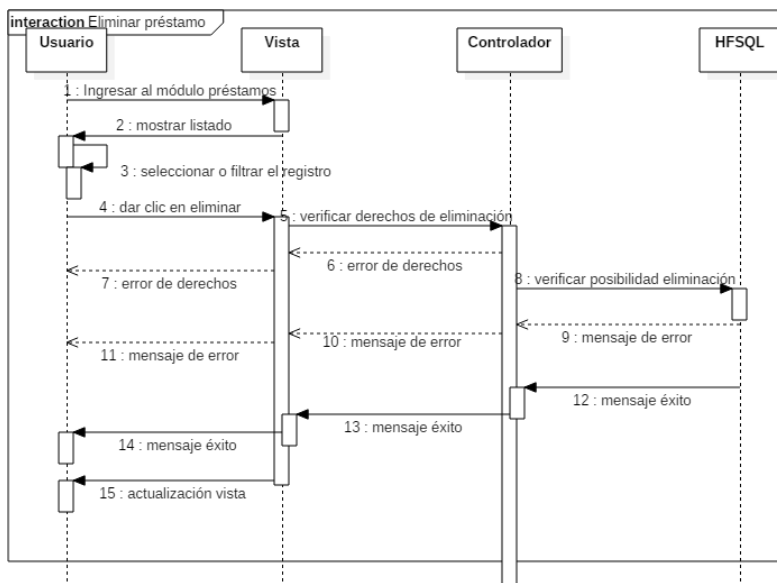
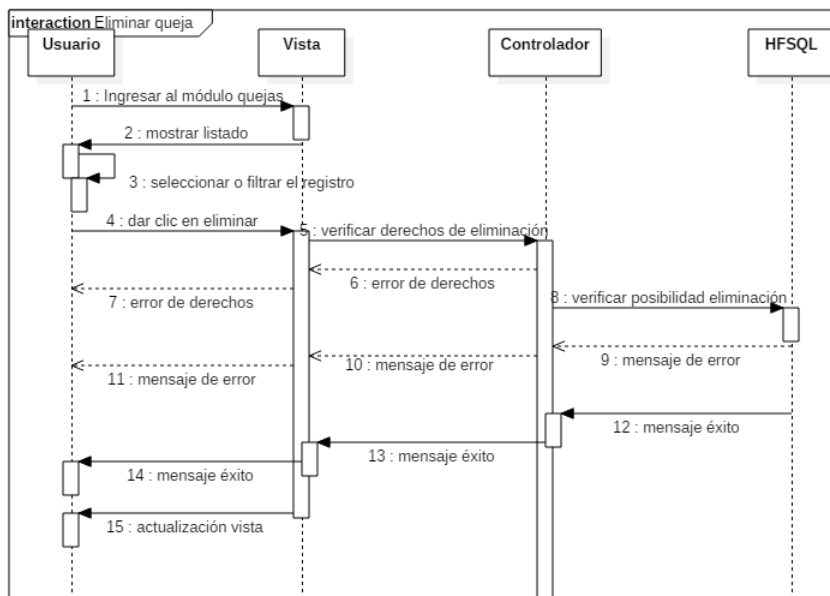


Ilustración 24: eliminar préstamo



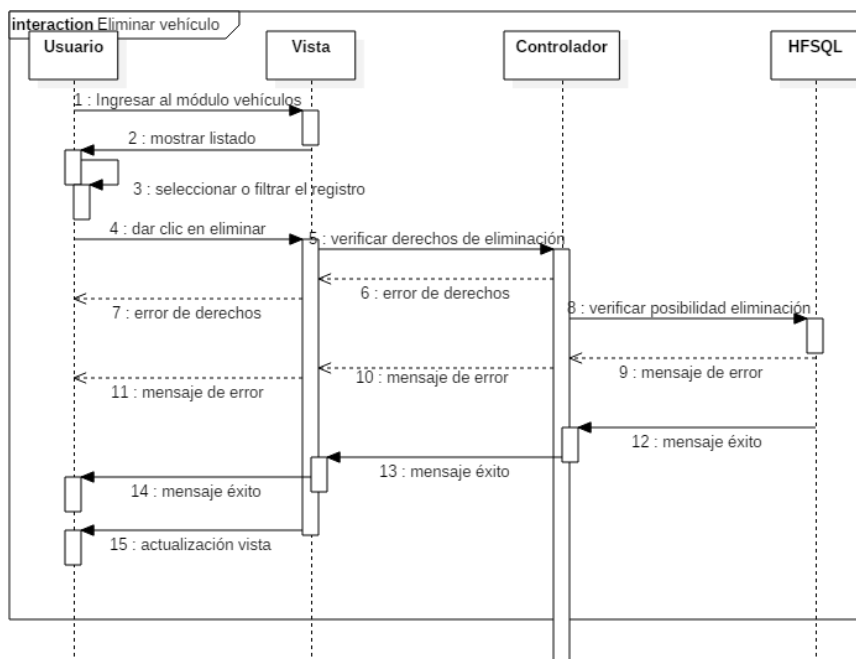
Ref: Autor

Ilustración 25: eliminar queja



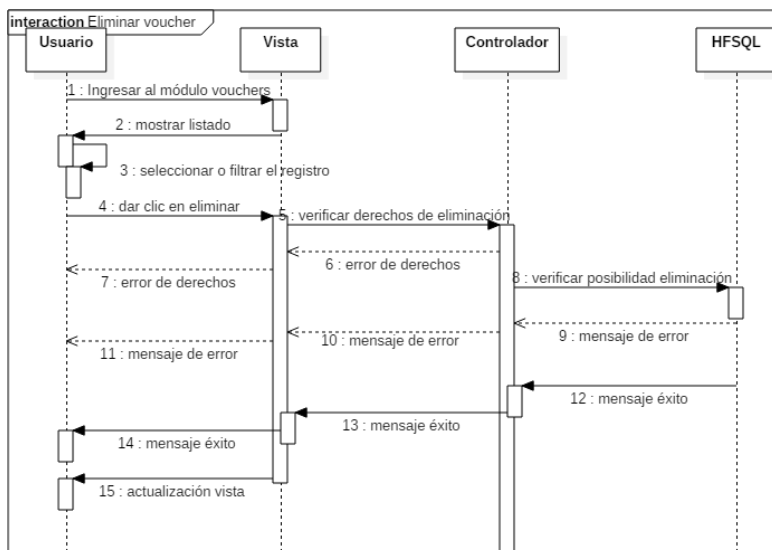
Ref: Autor

Ilustración 26: eliminar vehículo



Ref: Autor

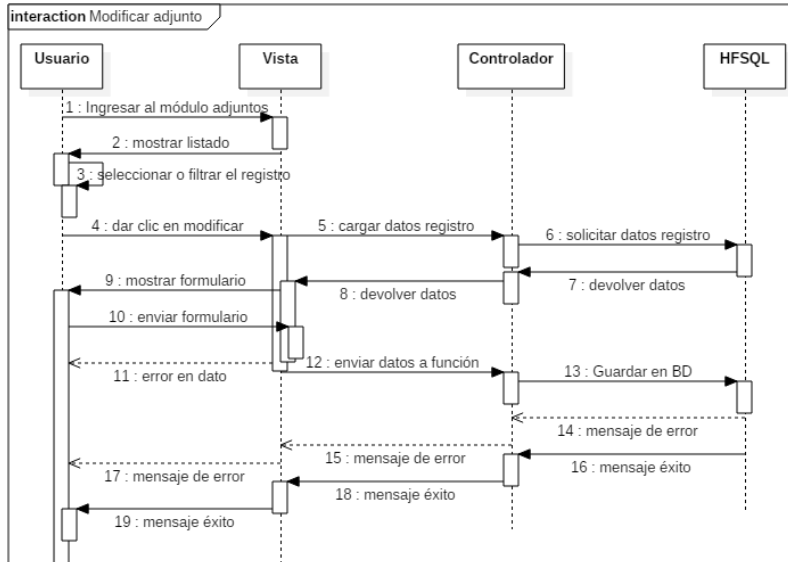
Ilustración 27: eliminar voucher



Ref: Autor

Modificar

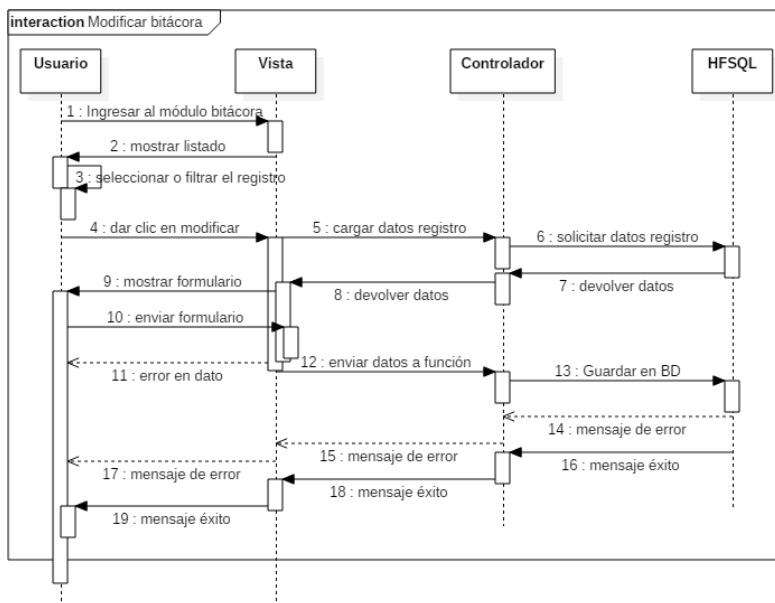
Ilustración 28: modificar adjunto



Ref: Autor

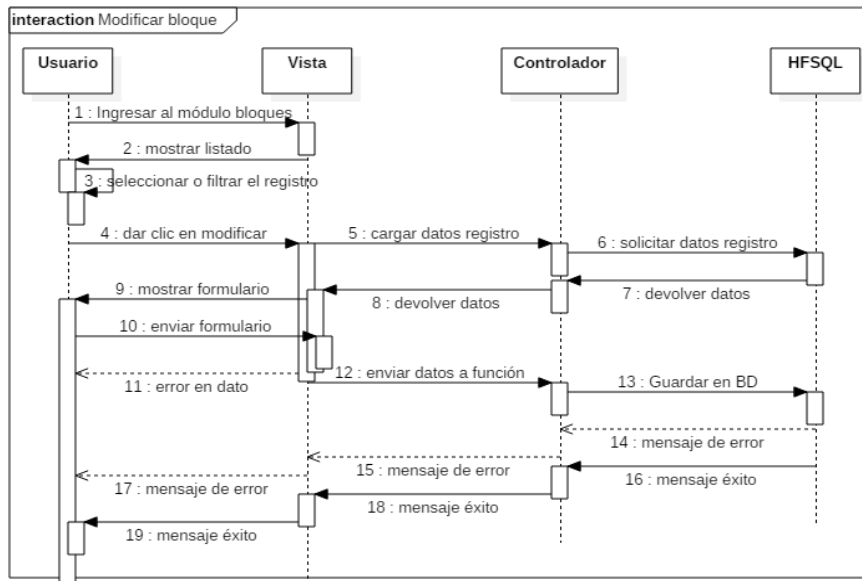
Bitácora

Ilustración 29: modificar bitácora



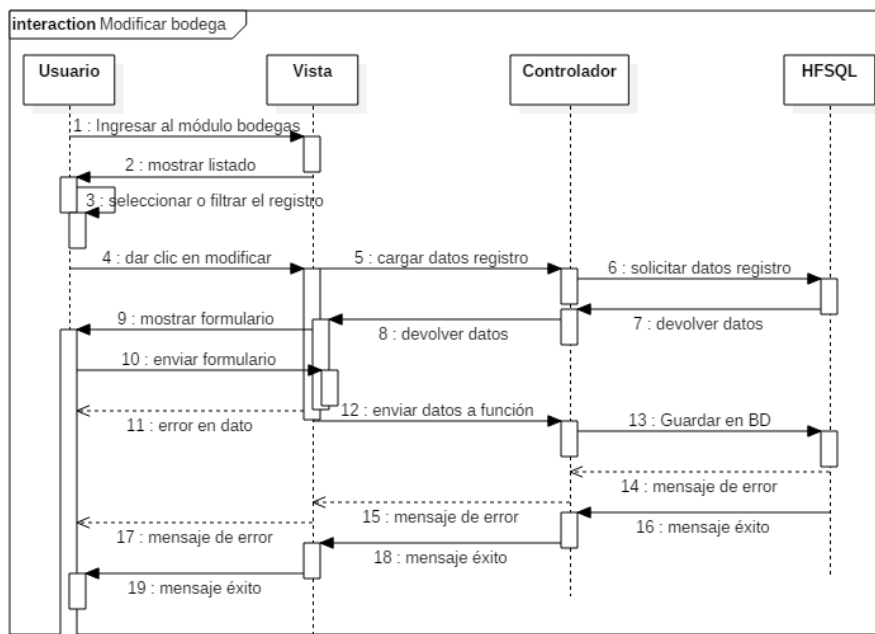
Ref: Autor

Ilustración 30: modificar bloque



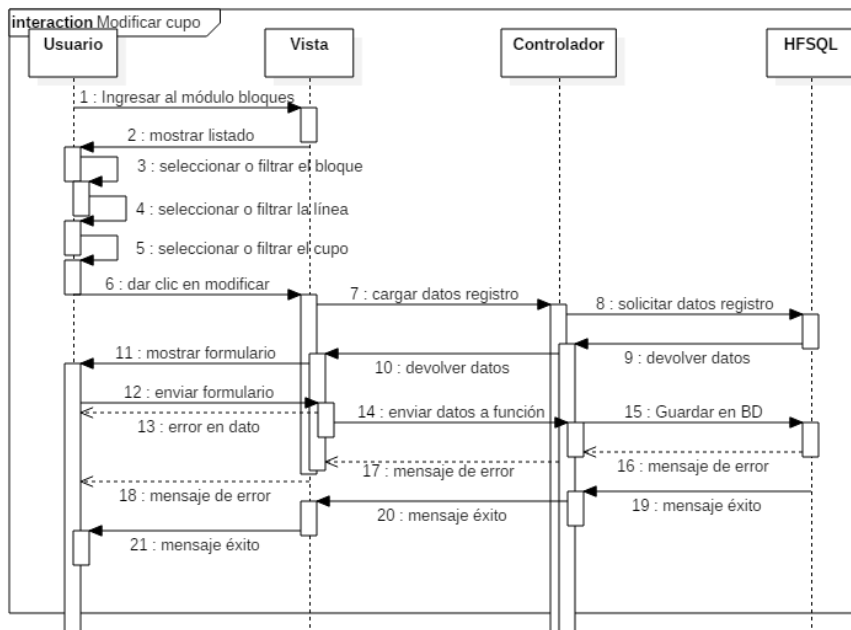
Ref: Autor

Bodega



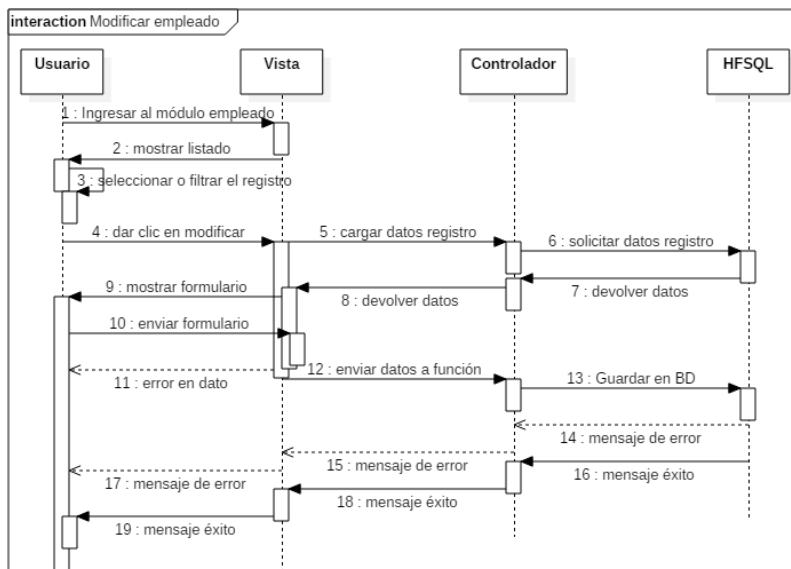
Ref: Autor

Ilustración 31: modificar cupo



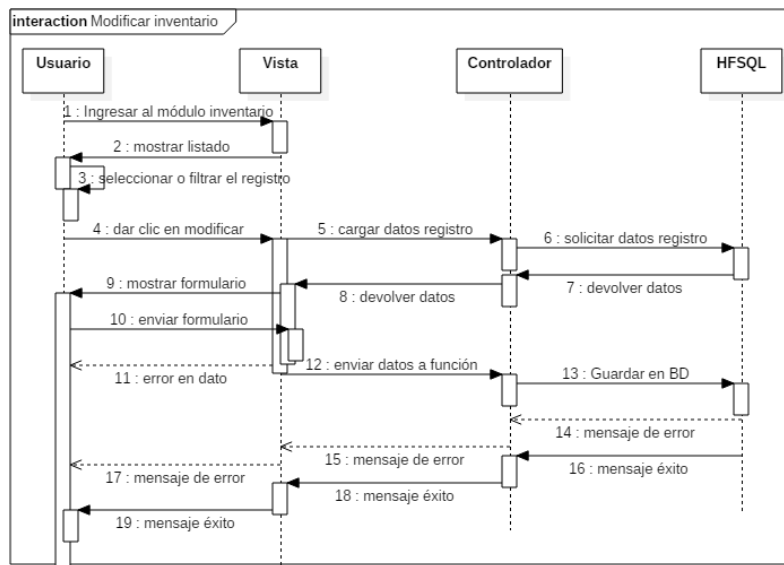
Ref: Autor

Ilustración 32: modificar empleado



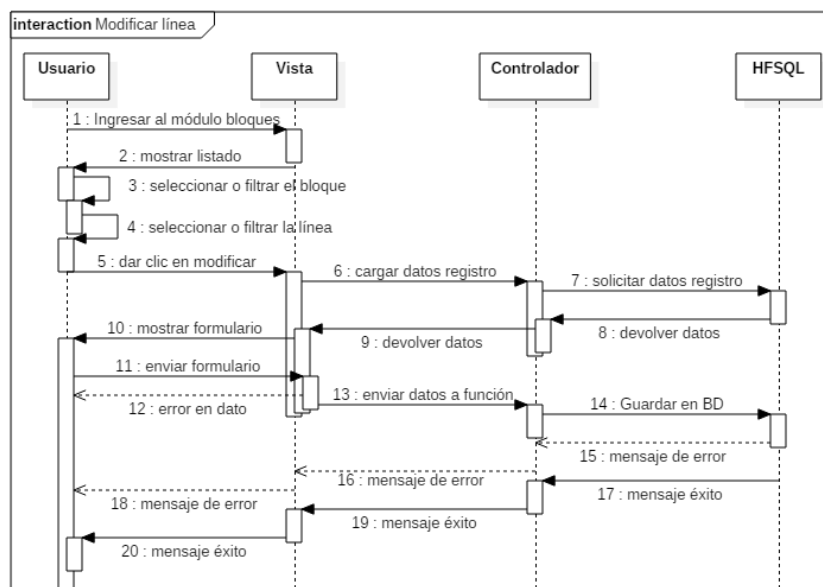
Ref: Autor

Ilustración 33: modificar inventario



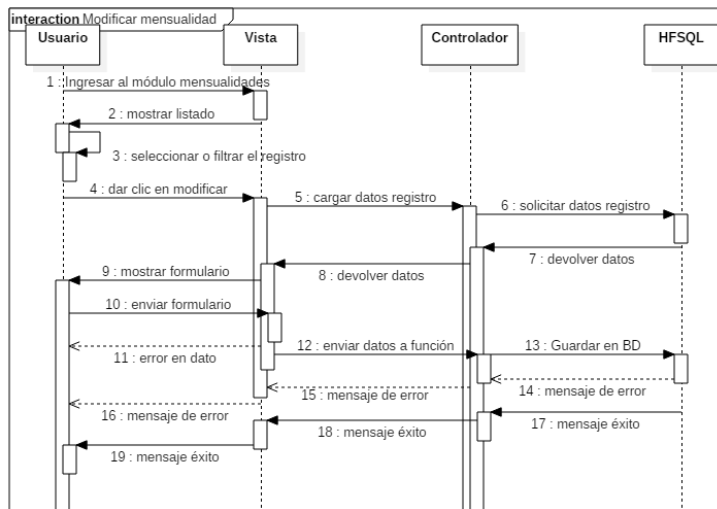
Ref: Autor

Ilustración 34: modificar línea



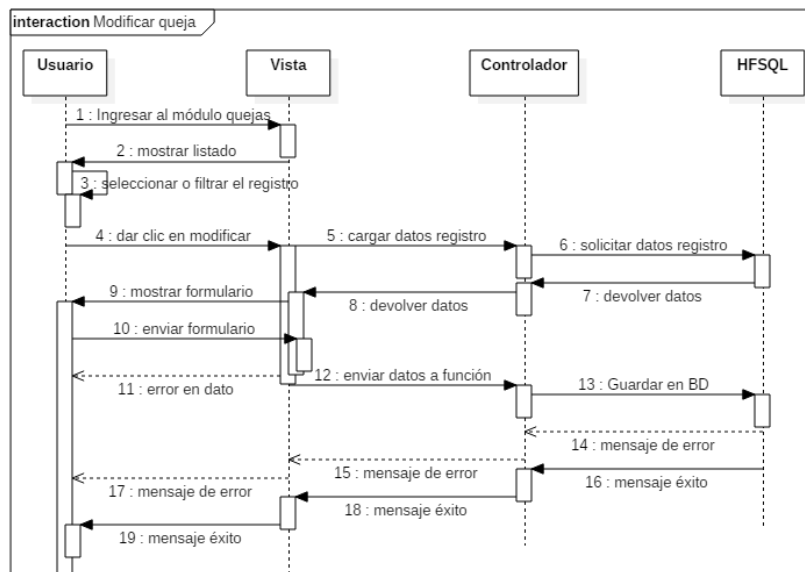
Ref: Autor

Ilustración 35: modificar mensualidad



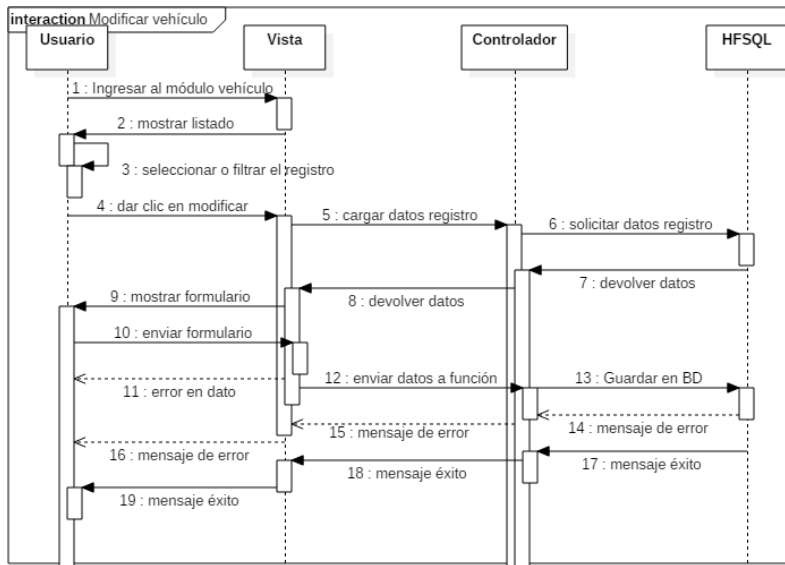
Ref: Autor

Ilustración 36: modificar queja



Ref: Autor

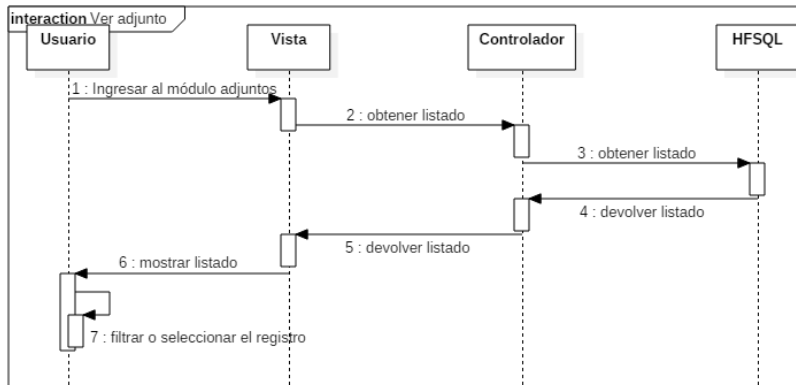
Ilustración 37: modificar vehículo



Ref: Autor

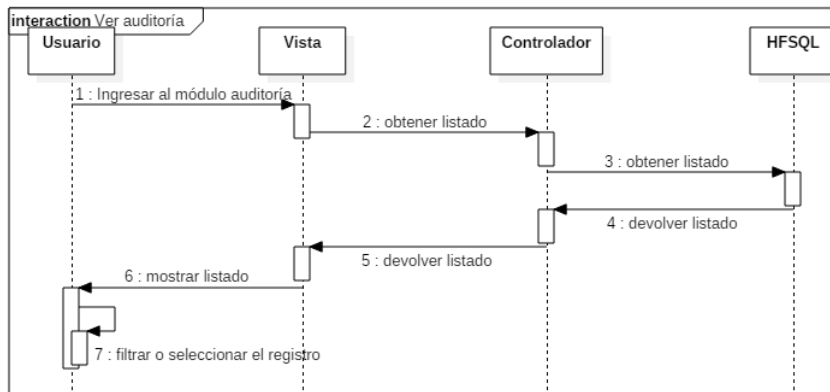
Ver

Ilustración 38: ver adjunto



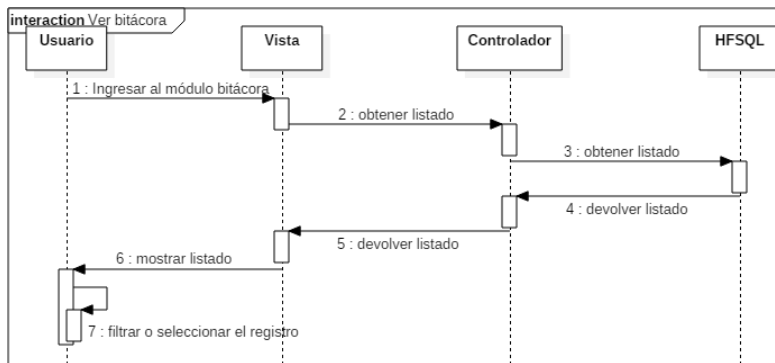
Ref: Autor

Ilustración 39: ver auditoría



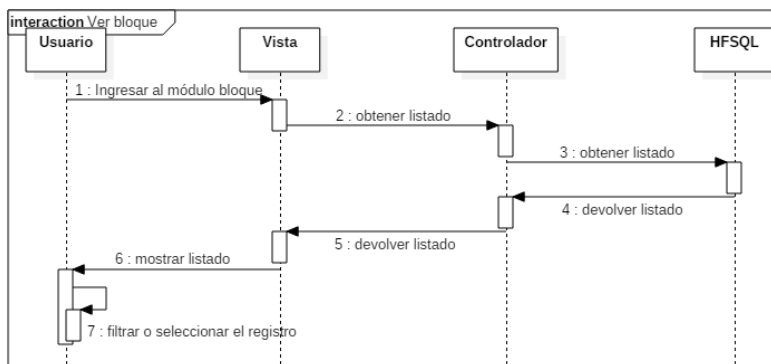
Ref: Autor

Ilustración 40: ver bitácora



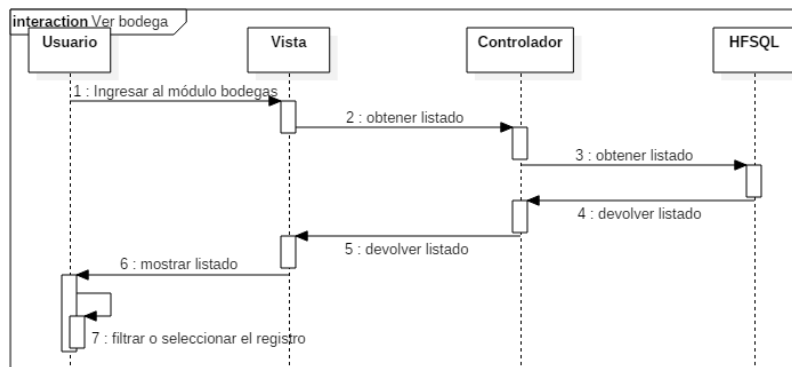
Ref: Autor

Ilustración 41: ver bloque



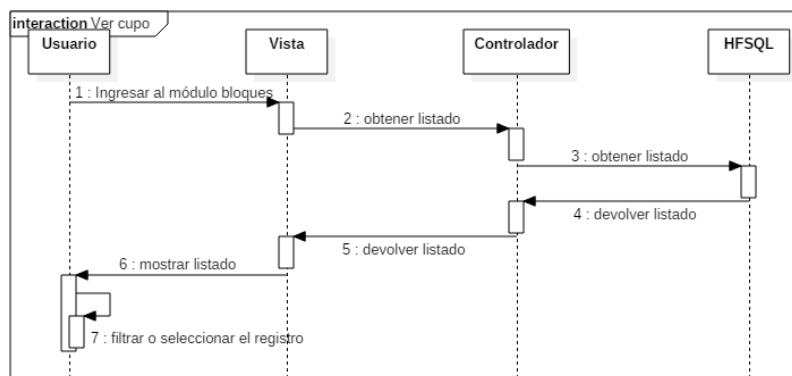
Ref: Autor

Ilustración 42: ver bodega



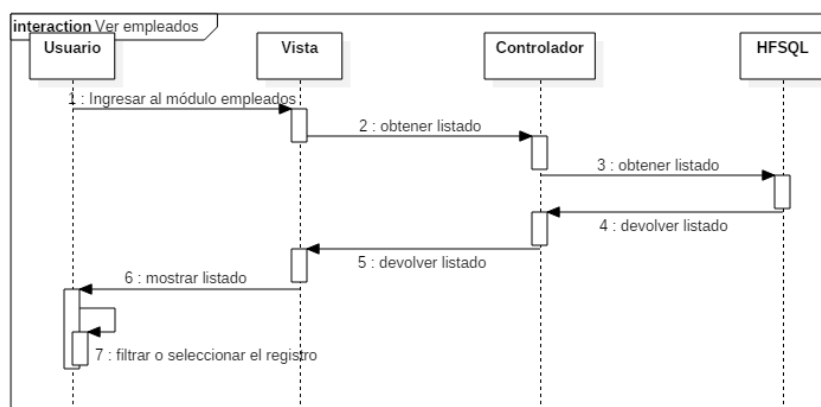
Ref: Autor

Ilustración 43: ver cupo



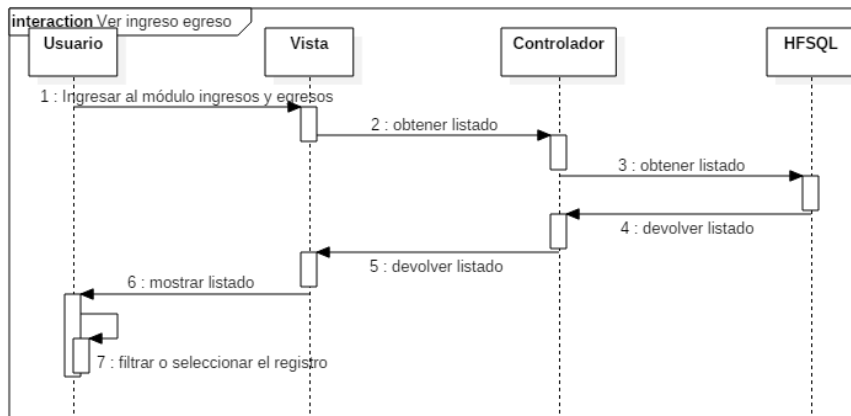
Ref: Autor

Ilustración 44: ver empleado



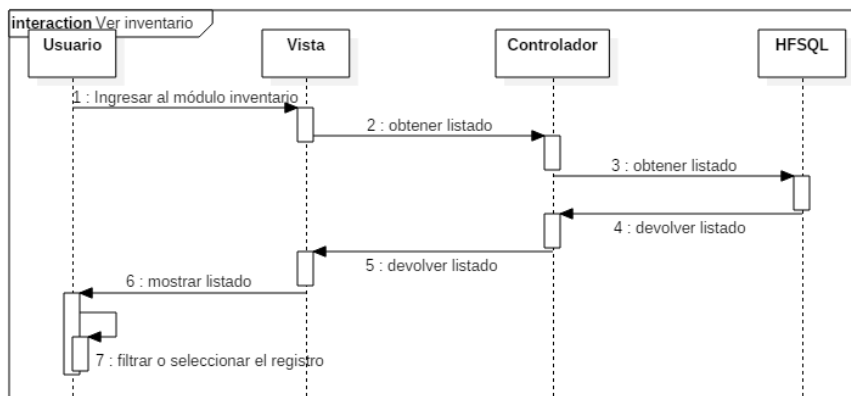
Ref: Autor

Ilustración 45: ver ingreso



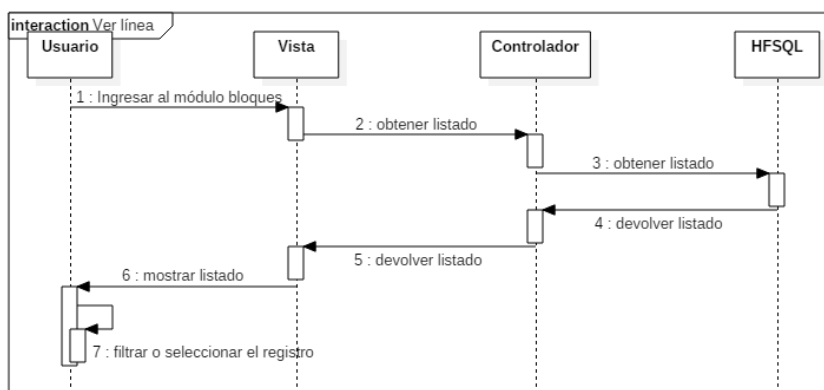
Ref: Autor

Ilustración 46: ver inventario



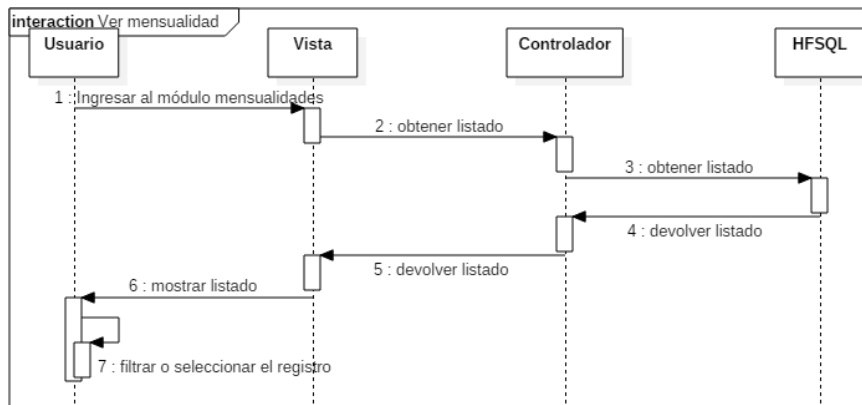
Ref: Autor

Ilustración 47: ver línea



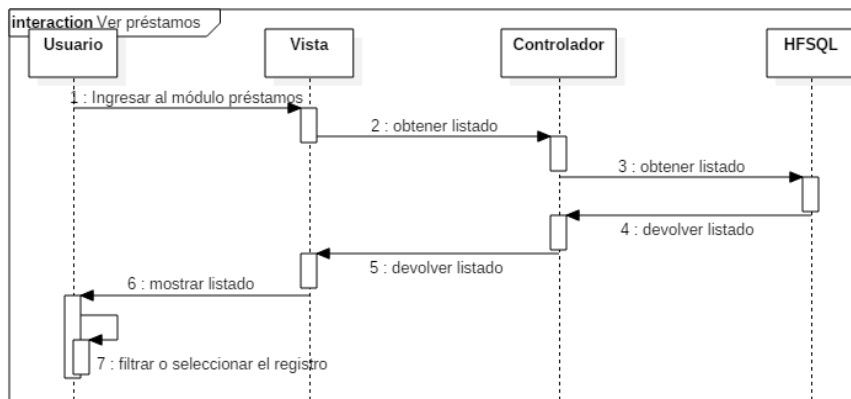
Ref: Autor

Ilustración 48: ver mensualidad



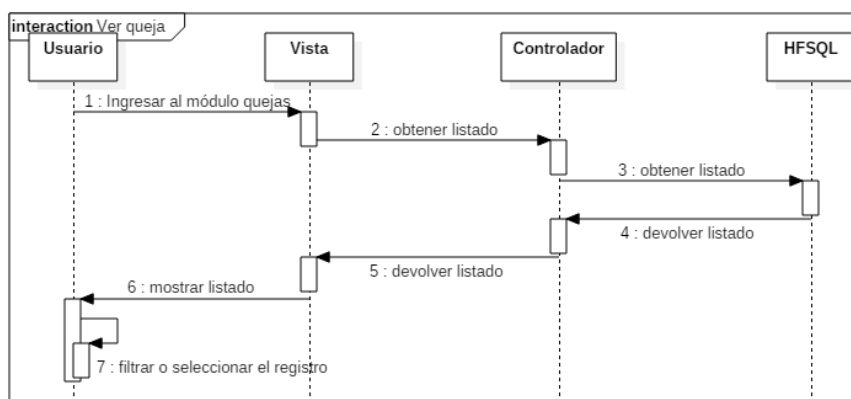
Ref: Autor

Ilustración 49: ver préstamo



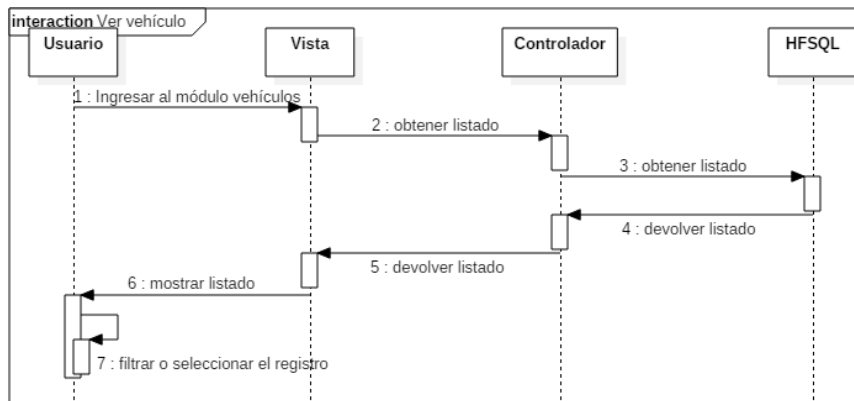
Ref: Autor

Ilustración 50: ver queja



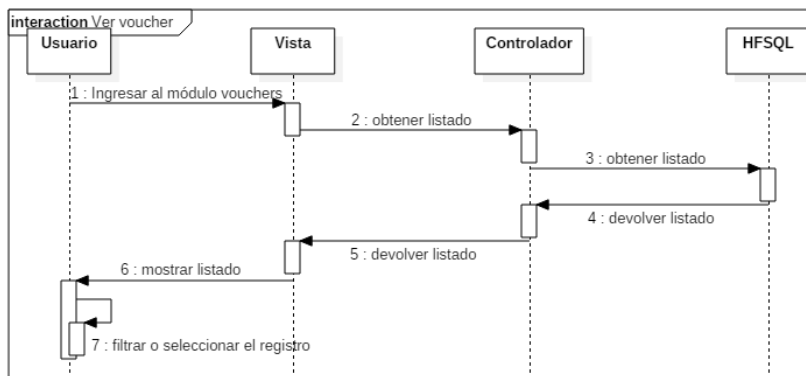
Ref: Autor

Ilustración 51: ver vehículo



Ref: Autor

Ilustración 52: ver voucher



Ref: Autor

Capítulo 5

Diseño del sistema

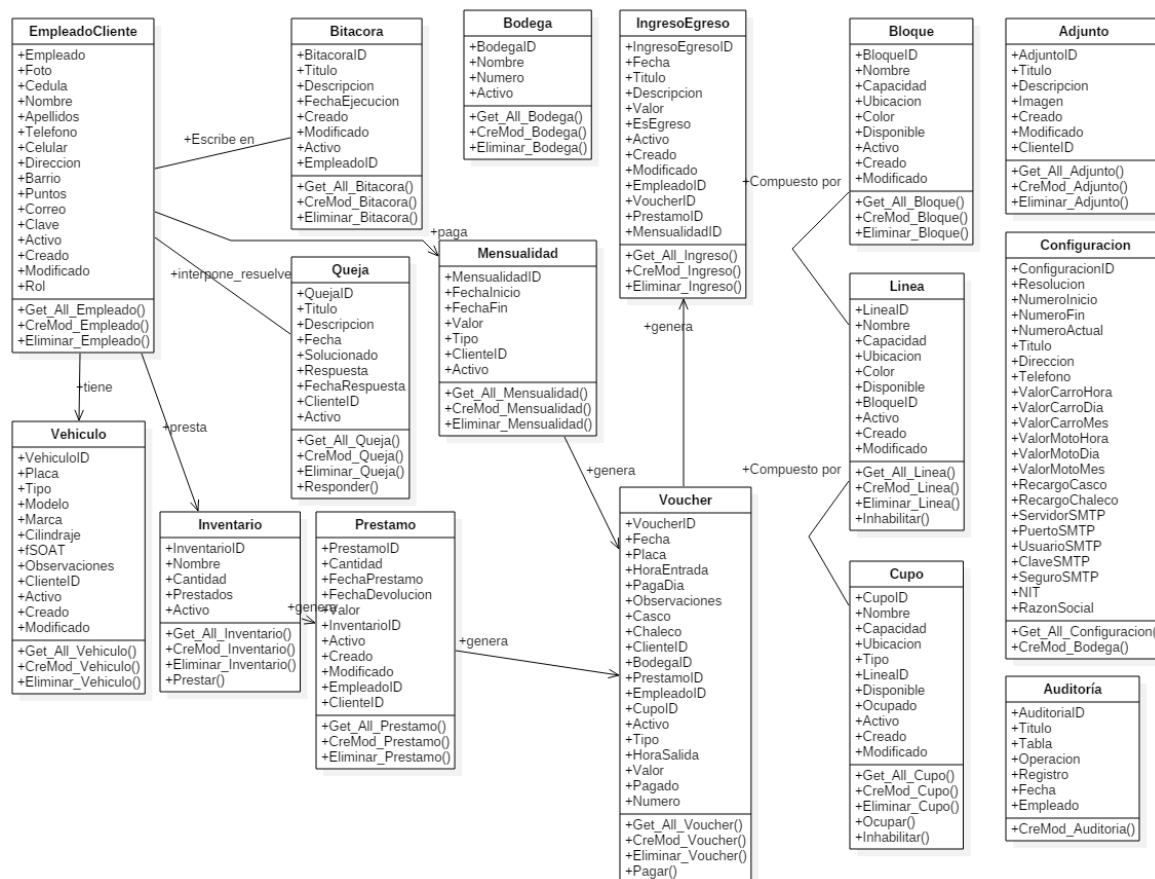
Diseño arquitectónico

El diseño arquitectónico que se eligió para el proyecto es del tipo Cliente-Servidor en donde existe un servidor centralizado que brinda los servicios necesarios para la interacción de los usuarios en el sistema y por otro lado están los clientes que sirven como interfaces de acceso a los servicios del servidor.¹¹

¹¹ INSTITUTO DE TECNOLOGÍAS EDUCATIVAS [<http://www.ite.educacion.es>]. 2016. Modelo Cliente-Servidor, [consulta 24 agosto 2016]. Disponible en: http://www.ite.educacion.es/formacion/materiales/157/cd/m1_1_conceptos_basicos_de_internet/modelo_clienteservidor.html.

Diagrama de clases

Ilustración 53: diagrama de clases



Ref: Autor

Tipos de función

En la anterior ilustración se pueden resaltar tres funciones principales que coinciden con la mayoría de clases; a continuación, se detalla su uso:

- **Get All x**

La función `Get_All`, es la función que se encarga de obtener los datos de la base de datos, y devolverlos en un arreglo de estructura, que tiene el mismo nombre de la clase. Esta función obtiene los datos a partir de consultas dinámicas, construidas a partir de los parámetros enviados.

- **CreMod x**

La función CreMod, es la función que se encarga de **crear** o **modificar** los datos, en la base de datos, devolviendo un **1** como respuesta de éxito, o un mensaje de error.

- **Eliminar_x**

La función Eliminar, es la función encargada de eliminar (lógicamente, en la mayoría de los casos) los registros de la base de datos.

Descripción de diagramas de clase

ADJUNTO

La clase adjunto es la clase que le permite al usuario subir registros adjuntos de facturas, documentos, fichas de inscripción y demás; en forma de imágenes, de modo que el usuario pueda realizar un diagnóstico o seguimiento sobre estos datos.

La clase **adjunto** posee los campos: id, título, descripción, imagen, creado, modificado, y clienteID

La clase **adjunto** posee los siguientes métodos

- **Get_All_Adjunto:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Adjunto:** es el método que permite crear o modificar adjuntos
- **Eliminar_Adjunto:** es el método que permite eliminar adjuntos de la base de datos

NOTAS: Los adjuntos sólo pueden ser eliminados por el usuario **administrador**.

AUDITORÍA

La clase auditoría es la clase que lleva el registro sobre cualquier operación de escritura en la base de datos, realizada por los empleados, o usuarios del software, de modo que se pueda llevar un seguimiento de responsabilidades sobre el sistema.

La clase **auditoría** posee los campos: id, título, tabla, operación, registro, fecha y empleadoID.

La clase **auditoría** posee los siguientes métodos

- **Get_All_Auditoría:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Auditoría:** es el método que permite crear registros de auditoría

NOTAS: Los registros de auditoría **no pueden ser modificados o eliminados** por ningún usuario.

BITÁCORA

La clase bitácora es la que permite la comunicación entre empleados a modo de anotaciones en libro, además del registro de novedades sobre el establecimiento. El empleado puede escribir mensajes, que los demás empleados verán al entrar al módulo

La clase **bitácora** posee los campos: id, título, fecha de ejecución, descripción, empleadoID, creado, modificado y activo.

La clase **bitácora** posee los siguientes métodos

- **Get_All_Bitacora:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Bitacora:** es el método que permite crear o modificar los registros de la bitácora
- **Eliminar_Bitacora:** es el método que permite eliminar las entradas de la bitácora, de la base de datos

NOTAS: La bitácora sólo puede ser **modificada** o **eliminada** por el usuario **administrador**.

BLOQUE

La clase bloque da una representación en el software, de los espacios físicos que compongan el establecimiento. Un bloque, generalmente representa un piso, destinado al parqueadero de vehículos.

La clase **bloque** posee los campos: id, nombre, capacidad, ubicación, color, disponible, activo, creado, modificado.

La clase **bloque** posee los siguientes métodos

- **Get_All_Bloque:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Bloque:** es el método que permite crear o modificar los registros
- **Eliminar_Bloque:** es el método que permite eliminar los bloques de la base de datos
- **Inhabilitar:** es el método que permite inhabilitar el bloque, en caso de daños u ocurrencias

NOTAS: Sin notas.

BODEGA

La clase bodega es la que permite crear bodegas en el software, para administrar las bodegas físicas, y el almacenamiento en las mismas.

La clase **bodega** posee los campos: id, nombre, número, y activo.

La clase **bodega** posee los siguientes métodos

- **Get_All_Bodega:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Bodega:** es el método que permite crear o modificar los registros
- **Eliminar_Bodega:** es el método que permite eliminar las bodegas de la base de datos

CLIENTE

La clase cliente es la que permite crear clientes, relacionarlos con vehículos, mensualidades, y tener un seguimiento de los mismos.

La clase **cliente** posee los campos: id, foto, cédula, nombre, apellidos, teléfono, celular, dirección, barrio, puntos, activo, creado, modificado, correo, y clave.

La clase **cliente** posee los siguientes métodos

- **Get_All_Cliente:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Cliente:** es el método que permite crear o modificar los registros
- **Eliminar_Cliente:** es el método que permite eliminar los clientes de la base de datos

NOTAS: los clientes sólo pueden ser **eliminados** por el usuario **administrador**.

CONFIGURACIÓN

La clase configuración es la que permite configurar todos los precios a usar en el sistema; hora, día o mes, de motos, carros; recargos por chaleco o casco y los parámetros SMTP para el envío de correos

La clase **configuración** posee los campos: id, resolución, número de inicio, número fin, número actual, título, dirección, teléfono, valor carro (hora, día, mes), valor moto (hora, día, mes), recargo casco recargo chaleco, servidor SMTP, puerto SMTP, usuario SMTP, clave SMTP, y servidor seguro SMTP; además del NIT y la razón social del establecimiento.

La clase **configuración** posee los siguientes métodos

- **Get_All_Configuración:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Configuración:** es el método que permite crear o modificar los registros
- **Eliminar_Configuración:** es el método que permite eliminar la configuración de la base de datos

NOTAS: la configuración sólo puede ser **modificada** por el usuario **administrador**.

CUPO

La clase cupo es la que permite crear lugares de parqueo dentro del sistema. Cada bloque posee una o más líneas, y cada línea posee uno o más cupos, que son donde el cliente dejará su vehículo.

La clase **cupo** posee los campos: id, nombre, capacidad, ubicación, tipo, disponible, ocupado, activo, creado, modificado, y líneaID.

La clase **cupo** posee los siguientes métodos

- **Get_All_Cupo:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Cupo:** es el método que permite crear o modificar los registros
- **Eliminar_Cupo:** es el método que permite eliminar los cupos de la base de datos

EMPLEADO

La clase empleado es la que permite administrar los datos de usuario de cada empleado, su contraseña, y la que habilita la auditoría.

La clase **empleado** posee los campos: id, foto, cédula, nombre, apellidos, teléfono, celular, dirección, barrio, puntos, activo, creado, modificado, correo, clave, y rol.

La clase **empleado** posee los siguientes métodos

- **Get_All_Empleado:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Empleado:** es el método que permite crear o modificar los registros
- **Eliminar_Empleado:** es el método que permite eliminar los empleados de la base de datos
- **RecuperarContrasena:** es el método que permite recuperar la contraseña de usuario

NOTAS: la **eliminación** de empleados sólo puede ser realizada por el usuario **administrador**. Sólo el usuario puede cambiar su propia contraseña.

INGRESO

La clase ingreso es la que permite llevar el registro de todos los ingresos y egresos del sistema, tanto de facturas, préstamos, mensualidades, como de ingresos y egresos creados manualmente.

La clase **ingreso** posee los campos: id, fecha, título, descripción, valor, esEgreso, activo, creado, modificado, empleadoID, voucherID, préstamoID, y mensualidadID.

La clase **ingreso** posee los siguientes métodos

- **Get_All_Ingreso:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Ingreso:** es el método que permite crear o modificar los registros
- **Eliminar_Ingreso:** es el método que permite eliminar los ingresos de la base de datos

NOTAS: la **eliminación** de ingresos o egresos, sólo puede ser realizada por el usuario **administrador**.

INVENTARIO

La clase inventario es la que permite crear ítems con fines de préstamo en la base de datos.

La clase **inventario** posee los campos: id, nombre, cantidad, prestados, y activo.

La clase **inventario** posee los siguientes métodos

- **Get_All_Inventario:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Inventario:** es el método que permite crear o modificar los registros
- **Eliminar_Inventario:** es el método que permite eliminar los ítems de la base de datos
- **Prestar:** es el método que aumenta o decrementa la cantidad de prestados

NOTAS: La **modificación** y **eliminación** de ítems sólo puede ser realizada por el usuario **administrador**.

LÍNEA

La clase línea es la que permite crear líneas de parqueo, dependientes de los bloques, las cuales contendrán cupos, que serán usados por los clientes.

La clase **línea** posee los campos: id, nombre, capacidad, ubicación, color, disponible, activo, creado, modificado, y bloqueID.

La clase **línea** posee los siguientes métodos

- **Get_All_Linea:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Linea:** es el método que permite crear o modificar los registros
- **Eliminar_Linea:** es el método que permite eliminar las líneas de la base de datos

NOTAS: para ser eliminada, una línea no debe tener cupos activos.

MENSUALIDAD

La clase mensualidad es la que permite crear mensualidades en el sistema, asignando las mismas a un cliente y verificando las mensualidades activas a la hora de facturar.

La clase **mensualidad** posee los campos: id, fecha inicio, fecha fin, valor, tipo, clienteID, y activo.

La clase **mensualidad** posee los siguientes métodos

- **Get_All_Mensualidad:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Mensualidad:** es el método que permite crear o modificar los registros
- **Eliminar_Mensualidad:** es el método que permite eliminar las mensualidades de la base de datos

- **Vencer:** procedimiento automático que verifica las mensualidades vencidas y las inhabilita

NOTAS: las mensualidades sólo pueden ser **modificadas** y **eliminadas** por el usuario administrador.

PRÉSTAMO

La clase préstamo es la que permite crear préstamos a partir del inventario actual del establecimiento.

La clase **préstamo** posee los campos: id, cantidad, fecha préstamo, fecha devolución, valor, activo, creado, modificado, inventarioID, empleadoID, y clienteID.

La clase **préstamo** posee los siguientes métodos

- **Get_All_Prestamo:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Prestamo:** es el método que permite crear o modificar los registros
- **Eliminar_Prestamo:** es el método que permite eliminar los préstamos de la base de datos
- **Pagar:** es el método que permite pagar el valor de un préstamo y darlo por cerrado.

NOTAS: los préstamos sólo pueden ser **modificados** o **eliminados** por el usuario administrador.

QUEJA

La clase queja es la que permite crear quejas sobre el servicio del establecimiento y darles respuesta.

La clase **queja** posee los campos: id, título, descripción, fecha, solucionado, respuesta, fecha respuesta, clienteID, y activo.

La clase **queja** posee los siguientes métodos

- **Get_All_Queja:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Queja:** es el método que permite crear o modificar los registros
- **Eliminar_Queja:** es el método que permite eliminar las quejas de la base de datos
- **Resolver:** es el método que permite dar respuestas a las quejas.

NOTAS: sin notas.

VEHÍCULO

La clase vehículo es la que permite crear vehículos y relacionarlos con los clientes. La clase **vehículo** posee los campos: id, placa, tipo, modelo, marca, cilindraje, fecha del soat, observaciones, activo, creado, modificado, y clienteID.

La clase **vehículo** posee los siguientes métodos

- **Get_All_Vehiculo:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Vehiculo:** es el método que permite crear o modificar los registros
- **Eliminar_Vehiculo:** es el método que permite eliminar los vehículos de la base de datos

NOTAS: la **modificación** y **eliminación** de vehículos, sólo puede ser realizada por el usuario **administrador**.

VOUCHER

La clase voucher es la que permite generar recibos (lógicos) en el sistema, para llevar el registro de servicios realizados.

La clase **voucher** posee los campos: id, fecha, placa, hora entrada, paga día, observaciones, casco, chaleco, clienteID, bodegaID, préstamoID, empleadoID, activo, cupoID, tipo, hora salida, valor, pagado, y número.

La clase **voucher** posee los siguientes métodos

- **Get_All_Voucher:** es el método que permite obtener los registros de la base de datos, con o sin filtros sobre sus campos
- **CreMod_Voucher:** es el método que permite crear o modificar los registros
- **Eliminar_Voucher:** es el método que permite eliminar los vouchers de la base de datos.
- **Pagar:** es el método que permite realizar pagos sobre los vouchers generados.

NOTAS: las **modificaciones** o **eliminaciones** de un voucher, sólo pueden ser realizadas por el usuario **administrador**.

Usuarios

Perfiles de usuario

Tabla 82: perfiles de usuarios

Tipo de usuario	Descripción	Funcionalidades
Administrador	Persona encargada de administrar los usuarios, parametrizar el sistema, y modificar y eliminar registros especiales	Leer, crear, modificar, eliminar cualquier registro. Configurar los precios Configurar los datos empresariales
Empleado	Usuario regular del sistema, con permisos básicos. No tiene permiso para modificar y eliminar registros especiales	Leer, crear, modificar, y eliminar registros básicos. Gestionar clientes, vehículos, mensualidades, quejas, ingresos, egresos y demás.

Ref: autor

Interfaces

A continuación, se muestra el diseño preliminar de pantallas usado para el desarrollo del software ParqueApp. Se muestra el diseño inicial y final para cada módulo.

Módulo de quejas

Ilustración 54: módulo de quejas

The image displays the initial and final design for the complaint module, including wireframes and mockups.

Wireframes:

- Enviar queja:** A form with fields for 'Fecha', 'Titulo', and a text area 'Escriba aquí su queja'. It includes 'Guardar' and 'Cancelar' buttons.
- Resolver:** A form with fields for 'Fecha', 'Titulo', a text area 'Escriba aquí su queja', and another text area 'Escriba aquí la respuesta a la queja'. It includes 'Guardar' and 'Cancelar' buttons.
- Listado de quejas:** A table listing complaints with columns for 'Fecha', 'Titulo', and 'Descripción'.

Mockups:

- Queja (Form):** A window titled 'Queja' with a date picker set to '09/08/2016', a 'Titulo' field, a 'Descripción' text area, and 'Guardar' and 'Cancelar' buttons.
- Queja (Table):** A window titled 'Queja' showing a table with columns: 'Fecha', 'Titulo', 'Descripción', 'Solucionado', 'Respuesta', and 'FechaRespuesta'. It includes 'Crear', 'Modificar', 'Resolver', and 'Eliminar' buttons at the bottom.

Fecha	Titulo	Descripción
15/03/2016	Mal servicio	hay mal servicio por esto y lo otro
15/03/2016	Mal servicio	hay mal servicio por esto y lo otro
15/03/2016	Mal servicio	hay mal servicio por esto y lo otro
15/03/2016	Mal servicio	hay mal servicio por esto y lo otro
15/03/2016	Mal servicio	hay mal servicio por esto y lo otro

Fecha	Titulo	Descripción	Solucionado	Respuesta	FechaRespuesta

Ref: Autor

Descripción de diseño de pantalla

El módulo de quejas es el que permite crear quejas de los clientes, así como una respuesta a la queja. De modo que se pueda tener un seguimiento de quejas del establecimiento.

Módulo de auditoría

Ilustración 55: módulo de auditoría

Fecha	Título	Tabla	Operación	Registro
15/03/2016	Crear Voucher	VOUCHER	Creación	1+23/07/2016
15/03/2016	Crear Voucher	VOUCHER	Creación	1+23/07/2016
15/03/2016	Crear Voucher	VOUCHER	Modificación	1+23/07/2016
15/03/2016	Crear Voucher	VOUCHER	Eliminación	1+23/07/2016
15/03/2016	Crear Voucher	VOUCHER	Creación	1+23/07/2016

The screenshot shows the 'Auditoría' application window. The title bar contains the text 'Auditoría' and standard window controls. Below the title bar, there is a header section with four input fields labeled 'Título', 'Tabla', 'Operación', and 'Registro'. Below the header is a large table with five columns: 'Título', 'Tabla', 'Operación', 'Registro', and 'Fecha'. The table has many empty rows, indicating it is ready for data entry. The 'Fecha' column has a small arrow icon next to it, suggesting a date picker.

Ref: Autor

Descripción de diseño de pantalla

El módulo de auditoría lleva un registro las operaciones de escritura sobre la base de datos, junto al usuario que realizó la operación.

Módulo de inventario

Ilustración 56: módulo de inventario

The image shows two representations of the inventory module: a schematic diagram and a screenshot of a software window.

Schematic Diagram:

- Crear inventario:** A form with three input fields labeled "Nombre", "Cantidad", and "Prestados". Below these fields are two buttons: "Guardar" and "Cancelar".
- Inventario:** A table with three columns: "Nombre", "Cantidad", and "Prestados". The table contains two rows of data:

Nombre	Cantidad	Prestados
Casco	7	2
Chaleco	10	6

Screenshot:

The screenshot shows a window titled "Inventario" with a standard Windows title bar. Inside the window, there are three input fields labeled "Nombre", "Cantidad", and "Prestados". The "Cantidad" and "Prestados" fields have the value "0" entered. Below the input fields are two buttons: "Guardar" and "Cancelar".

Ref: Autor

Descripción de diseño de pantalla

El módulo de inventario permite crear elementos que pueden ser dados en préstamo a un cliente, por un periodo de tiempo y un valor.

Módulo de bitácora

Ilustración 57: módulo de bitácora

The image illustrates the Bitácora module through a diagram and two screenshots.

Diagram:

- Crear entrada:** A form with fields for 'Fecha' (Date), 'Título' (Title), and a large text area for 'Escriba aquí la entrada' (Write here the entry). It includes 'Guardar' (Save) and 'Cancelar' (Cancel) buttons.
- Bitácora:** A table with three columns: 'Fecha', 'Título', and 'Descripción'. It contains two entries:

Fecha	Título	Descripción
15/03/2016	Llamar	Llamar a don Manuel
15/03/2016	Observación	Hay que limpiar el estante #3

Screenshots:

- The top screenshot shows the Bitácora window with a table that includes an 'Empleado' (Employee) column. The table has headers for 'Fecha', 'Título', 'Descripción', and 'Empleado'. Below the table are buttons for 'Crear' (Create), 'Modificar' (Modify), and 'Eliminar' (Delete).
- The bottom screenshot shows the 'Crear entrada' form with the following data:
 - Fecha: 09/08/2016
 - Título: Bitácora 20160809
 - Descripción: (empty text area)Buttons for 'Guardar' (Save) and 'Cancelar' (Cancel) are at the bottom.

Ref: Autor

Descripción de diseño de pantalla

El módulo de bitácora permite crear entradas, peticiones, notas, gestiones y demás; que pueden ser vistas por los empleados.

Módulo de bodega

Ilustración 58: módulo de bodega

The diagram illustrates the warehouse module interface, consisting of three main components:

- Crear estante (Create Shelf):** A form with three input fields labeled "Número", "Capacidad", and "Usados". Below the fields are two buttons: "Guardar" (Save) and "Cancelar" (Cancel).
- Estanterías (Shelves):** A table displaying shelf information. The table has three columns: "Número", "Capacidad", and "Usados". The data shown is as follows:

Número	Capacidad	Usados
1	7	2
2	10	6
- Bodegas (Warehouses):** A main window titled "Bodegas" with a table for listing warehouses. The table has two columns: "Número" and "Nombre". At the bottom of the window are three buttons: "Crear" (Create), "Modificar" (Modify), and "Eliminar" (Delete). A modal dialog titled "Bodega" is shown in the foreground, containing two input fields: "Nombre" and "Número", and two buttons: "Guardar" (Save) and "Cancelar" (Cancel).

Ref: Autor

Descripción de diseño de pantalla

El módulo de bodega permite crear lugares donde se guardarán elementos como cascos, chalecos y otros; de modo que se sepa exactamente dónde está un elemento de una factura.

Módulo de clientes

Ilustración 60: módulo de clientes

The image displays three screenshots of a software interface for managing clients, titled 'Cliente'.

The top-left screenshot shows a form for adding or editing a client. It includes fields for Cédula, Nombre, Apellidos, Teléfono, Celular, Dirección, Barrio, Puntos, Departamento, Ciudad, and Correo. There are also buttons for 'Guardar' (Save) and 'Cancelar' (Cancel).

The top-right screenshot shows a similar form, but with a 'Foto' field and a 'Puntos' field. It also has 'Guardar' and 'Cancelar' buttons.

The bottom screenshot shows a table with columns for Foto, Cédula, Nombre, Apellidos, Teléfono, and Celular. Below the table are buttons for 'Crear' (Create), 'Modificar' (Modify), 'Crear Vehículo' (Create Vehicle), 'Pagar mensualidad' (Pay monthly fee), and 'Eliminar' (Delete).

Ref: Autor

Descripción de diseño de pantalla

El módulo de clientes/empleados permite gestionar los mismos dentro del software; llevar las mensualidades, vehículos, quejas de los clientes; y las operaciones internas de los empleados.

Módulo de vehículos

Ilustración 61: módulo de vehículos

The image displays three screenshots of a vehicle management software interface.

The top left screenshot shows a vertical form titled "Vehículo" with the following fields: Placa, Tipo (dropdown), Marca, Modelo, Cilindraje, and f venc SOAT. At the bottom are buttons for "Guardar" and "Cancelar".

The top right screenshot shows a horizontal form titled "Vehículo" with the following fields: Cedula, Nombre, Placa, Tipo (dropdown), Modelo, Marca, cilindraje, fSOAT, and Nota. At the bottom are buttons for "Guardar" and "Cancelar".

The bottom screenshot shows a table titled "VehículoL" with the following columns: Placa, Tipo, Modelo, Marca, cilindraje, fSOAT, Observaciones, and Cliente. The table is empty. At the bottom are buttons for "Crear", "Modificar", and "Eliminar".

Ref: Autor

Descripción de diseño de pantalla

El módulo de vehículos permite gestionar los vehículos de un cliente.

Módulo de bloques

Ilustración 62: módulo de bloques

Bloque

Nombre

Capacidad

Ubicación

Color

☐ Disponible

Guardar

Cancelar

Línea	Capacidad
1	30
2	25

Línea

Nombre

Capacidad

Ubicación

Color

☐ Disponible

Guardar

Cancelar

Cupo	Tipo
1	Moto
2	Carro

Cupo

Nombre

Capacidad

Ubicación

Color

Moto

☐ Disponible

Guardar

Cancelar

BloqueL

Nombre

Nombre	Capacidad	Ubicación	Color	Disponible

Crear

Modificar

Eliminar

Nombre

Nombre	Capacidad	Ubicación	Color	Disponible

Crear

Modificar

Eliminar

Nombre

Nombre	Capacidad	Ubicación	Tipo	Disponible	Ocupado

Crear

Modificar

Eliminar

Bloque

Nombre

Capacidad

Ubicación

Color

☒ Disponible

Guardar

Cancelar

Línea

Nombre

Capacidad

Ubicación

Color

☒ Disponible

Guardar

Cancelar

Cupo

Nombre

Capacidad

Ubicación

Tipo

☒ Disponible
☐ Ocupado

Guardar

Cancelar

Ref: Autor

115

Descripción de diseño de pantalla

El módulo de aparcamientos permite crear pisos, bloques, líneas y unidades de parqueo.

Módulo de préstamos

Ilustración 63: módulo de préstamos

La imagen muestra dos interfaces de usuario para el módulo de préstamos. La ventana superior, titulada "Préstamo", es un formulario para registrar un préstamo. Incluye campos para "Placa", "Cedula", "Nombre", "Fecha" (con un icono de calendario), "Artículo" (una lista desplegable que muestra "Artículo"), "Cantidad" (con el valor "0") y "Valor" (con el valor "\$ 0,00"). Hay botones "Guardar" y "Cancelar". La ventana inferior, titulada "PrestamoL", es una pantalla de consulta o edición que muestra una tabla con las siguientes columnas: "F Préstamo", "Artículo", "Cantidad", "Valor" y "F Devolución". La tabla está vacía. En la parte inferior de esta ventana hay botones "Crear", "Modificar" y "Eliminar".

Ref: Autor

Descripción de diseño de pantalla

El módulo de préstamos permite que se realicen préstamos de inventario a los clientes registrados en el sistema; por un periodo y un valor.

Módulo de mensualidades

Ilustración 64: módulo de mensualidades

The screenshot displays the 'Mensualidad' application interface. On the left, a vertical sidebar contains buttons for 'Cédula cliente', 'Nombre cliente', 'Fecha Inicio', 'Fecha Fin', 'Valor', 'Tipo', 'Guardar', and 'Cancelar'. The main window, titled 'Mensualidad', contains a form with the same fields: 'Cedula', 'Nombre', 'FechaInicio', 'FechaFin', 'Tipo', and 'Valor'. The 'Valor' field is pre-filled with '\$ 0,00'. Below the form are 'Guardar' and 'Cancelar' buttons. At the bottom of the application window is a table with the following columns: 'FechaInicio', 'FechaFin', 'Tipo', 'Valor', and 'Cliente'. The table is currently empty. At the bottom of the window are three buttons: 'Crear', 'Modificar', and 'Eliminar'.

Ref: Autor

Descripción de diseño de pantalla

El módulo de mensualidades permite crear mensualidades para un cliente, por un periodo y por un valor.

Módulo de facturas

Ilustración 65: módulo de facturas

[illegible]

Ref: Autor

Descripción de diseño de pantalla

El módulo de vouchers/facturas permite crear facturas de servicio de parqueadero en el sistema.

Capítulo 6

Conclusiones

- A través del proyecto realizado se logró crear un software que permite la administración de un establecimiento de alquiler de parqueaderos, haciendo las actividades ejecutadas más simples y rápidas.
- La recolección de datos se llevó a cabo por medio de entrevistas y observaciones a algunos parqueaderos de la ciudad de Pereira, separados geográficamente más de 100 metros.
- Como resultado de la investigación estadística se concluye que la plataforma más utilizada en estos establecimientos se trata de equipos portátiles con sistema operativo Windows 7.
- A través de las historias de usuario se definieron las características que debía tener el software, para satisfacer las necesidades de los administradores y empleados; siendo los requerimientos levantados a través de estas historias de usuario.
- A través de las metodologías, técnicas, y herramientas seleccionadas, se logró realizar el análisis, y diseño del software, cumpliendo con los requerimientos levantados, dando estabilidad y escalabilidad a la solución.
- La realización del proyecto dio a conocer algunas metodologías, técnicas y herramientas que servirán como base para la realización de futuros proyectos.

Recomendaciones

- Una segunda etapa del software, comprende el desarrollo del aplicativo en un ambiente WEB, para que su distribución pueda ser realizada como SaaS (software as a service).
- Se deben desarrollar web services o una API Rest, basada en el software web, que permita escalar el software a una plataforma móvil.
- Una tercera etapa del software, comprende el desarrollo de una aplicación móvil conectada al sistema principal.
- Se debe desarrollar un sistema de procesamiento de imágenes, para la lectura de placas a través de dispositivos móviles.

Director de trabajo

PhD. Guillermo Roberto Solarte

Colaboradores

- Guillermo Roberto Solarte Martínez
- Víctor Javier Valencia
- Ana Lucía Correa Arias
- Foro WINDEV

Bibliografía

Capacitación

- [1] COMPUTER ORG [www.computer.org]. 2016. SWEBOK, [consulta 24 agosto 2016]. Disponible en: <http://www.computer.org/web/swebok>.
- [2] Daniel H. Steinberg and Daniel W. Palmer: Extreme Software Engineering, Pearson Education, Inc., ISBN 0-13-047381-2
- [3] DIARIO LA REPÚBLICA [www.larepublica.co]. 2016. Secretaría tiene bajo la lupa 3000 parqueaderos de Bogotá, [consulta 24 agosto 2016]. Disponible en: http://www.larepublica.co/secretar%C3%ADa-tiene-bajo-la-lupa-3000-parqueaderos-de-bogot%C3%A1_211946.
- [4] HITSLINK NET APPLICATIONS [www.hitslink.com]. 2016. Desktop Operating System Market Share, [consulta 24 Agosto 2016]. Disponible en: <http://marketshare.hitslink.com/operating-system-market-share.aspx?qprid=10&qpcustomd=0&qpsp=175&qnp=1&qptimeframe=M>
- [5] MICROSOFT MSDN [msdn.microsoft.com]. 2016. Model-View-Controller, [consulta 24 agosto 2016]. Disponible en: <https://msdn.microsoft.com/en-us/library/ff649643.aspx>
- [6] WINDEV [www.windev.com]. 2016. WEBDEV TUTORIAL, [consulta 24 agosto 2016]. Disponible en: <http://www.pcsoft-windev-webdev.com/tutorial/us/tutorial-webdev-19.pdf>
- [7] STATCOUNTER [www.statcounter.com]. 2016. Top 7 Desktop & Console Oss from 5 to 7 Apr 2014, [consulta 24 agosto 2016]. Disponible en: <http://gs.statcounter.com/#os-www-daily-20140405-20140407-bar>
- [8] W3COUNTER [www.w3counter.com]. 2016. Browser & Platform Market Share, [consulta 24 Agosto 2016]. Disponible en: <https://www.w3counter.com/globalstats.php?a%C3%B1o=2014&mes=3>
- [9] WINDEV [www.windev.com]. 2016. Windev Development Help, [consulta 24 Agosto 2016]. Disponible en: <http://help.windev.com>.
- [10] WINDEV [www.windev.com]. 2016. Windev Development Forum, [consulta 24 Agosto 2016]. Disponible en: <http://forum.windev.com>.

- [11] WINDEV ONLINE [www.windevonline.com]. 2016. Videos Windev Online, [consulta 24 agosto 2016]. Disponible en: <http://www.windevonline.com>.
- [12] WIKIPEDIA [www.wikipedia.com]. 2016. Versiones de Windows más utilizadas, [consulta 24 agosto 2016]. Disponible en: https://es.wikipedia.org/wiki/Microsoft_Windows.
- [13] INSTITUTO DE TECNOLOGÍAS EDUCATIVAS [http://www.ite.educacion.es]. 2016. Modelo Cliente-Servidor, [consulta 24 agosto 2016]. Disponible en: http://www.ite.educacion.es/formacion/materiales/157/cd/m1_1_conceptos_basicos_de_internet/modelo_clienteservidor.html.